EVALUATING SYSTEMS OF SYSTEMS AGAINST MISSION REQUIREMENTS

by

Philip Johnson, MEng (Hons) DIS

A Doctoral Thesis submitted in partial fulfilment of the requirements for the award of
Doctor of Philosophy of Loughborough University

October 2009

Preface

[redacted]

Dr Emmett Brown, Back to the Future

In late 2005 I was interviewed at Loughborough to undertake a PhD with Professor Mike Woodhead and Dr Carys Siemieniuch and was subsequently offered the studentship. To my parent's delight I left a well paid job in industry to return to the university where I had spent the previous five years studying for my Masters degree. I jest, they were very supportive. As I write this my PhD time is about to come to an end and, at twenty seven years of age, I find myself still a student and in the unusual position of having avoided income tax for the majority of my adult life. On this point alone I consider my PhD to have been a great success. Sadly tax avoidance is not the criteria on which a PhD is awarded, rather it is the thesis that lies beneath this page.

This thesis is a record of my research efforts over the past three years; essentially it is a story of what I have done... a non-fictional and academically rigorous story I hasten to add. The research problem that this thesis addresses required a number of different research threads to answer it. Through hard work and a (substantial) dash of luck these research threads combined at the end of this research into a new approach to tackling system of systems problems. With the uncertainties present in any research activity the chances of this occurring has, at times, felt on par with the quote at the start of this preface by Dr Emmett Brown. The output of this is not an end point, but rather a natural stopping point for this research given the three year PhD time constraint. The journey described herein establishes the current state of the art and then advances us a small step while providing indicators of future applications and further research. I have been very lucky to be able to pursue threads of research which have interested and entertained me. After months of writing, revisions and rewrites I hope this thesis still conveys that passion.

Phil Johnson

October 2009

Abstract

This thesis investigates the nature of systems problems and the need for an open viewpoint to explain a system by viewing it as part of a larger whole and explaining its role in terms of that larger whole. The problem this research investigates is wicked and hence is unique in each instance. Therefore, an empirical proof would only hold for that particular instantiation of the problem, not the problem as a whole. After exposing some of the limitations of traditional systems engineering to this type of problem it is clear that a new approach is needed. The approach taken in the thesis is model driven and it is the architecture of this approach that is the stable artefact rather than the artefacts of a particular solution. The approach developed in this research has been demonstrated to be practicable.

Specifically, this research has developed and demonstrated a novel approach for a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission. Such planning decisions are wicked due to the uncertain and unique nature of military missions. Critical rationalism was used to validate the model driven approach and to falsify a parametric approach representative of traditional systems engineering through historical case studies. The main issue found with the parametric approach was the entanglement of functionality with the individual systems selected to implement the system of systems. The advantage of the model driven approach is that it separates functionality from implementation and uses model transformation for systems specification.

Thus, although wicked problems do not have an exhaustively describable set of potential solutions this thesis has shown that they are not unapproachable.

Table of Contents

List of Illustrations

List of Tables

List of Matrices

Acknowledgements

First of all I would like to give a special note of acknowledgement to Professor Mike Woodhead with whom I started this research and was my Supervisor until he unexpectedly took early retirement due to his health. Shortly afterwards he passed away in late 2007. Mike's guidance and input was instrumental during the first year of my PhD.

I would like to thank my Director of Research, Dr Carys Siemieniuch, who has seen me all the way through the past four years. Her advice and guidance has been key to moving my research from initial concept to completion. This journey has been far more eventful than the norm (if there is such a thing for PhD research) and that I have got to this stage is testament to her abilities.

My Supervisor, Professor Charles Dickerson, deserves special acknowledgement as he has shaped and directed the research contained in this thesis. To take over as a Supervisor more than half way through a student's PhD and help guide it towards completion is no mean feat. For the guidance and for the experience, thank you.

That this thesis may be considered an academic work at all is due to the advice and guidance of my academic team acknowledged above.

That I have actually completed this thesis is thanks to those who have offered me support throughout my research time. To my mum, dad and sister, thank you. There is one final person, without whom I would never have done this, principally because when she decided to get a doctorate I knew I would need to get one as well... to my girlfriend, you know.

Finally, all that there is left for me to say before my thesis starts in earnest is that any mistakes remaining in this thesis are my responsibility and mine alone. Not that there are any mistakes of course...

This research was funded by the Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC), established by the UK Ministry of Defence (UK MoD), and contributes to SEAS DTC Project SER006 entitled Impact of Different Cultural Value Sets on Semi/Autonomous Systems Mission-based Decision Making.

Authors Declaration

This is to certify that I am responsible for the work submitted in this thesis, that the original work is my own except as specified in acknowledgements or in footnotes, and that neither the thesis nor the original work contained therein has been submitted to this or any other institution for a higher degree.

Signed:

(P.Johnson)

Chapter 1
Motivation

Outline of Chapter

This chapter introduces the research problem that this thesis addresses. The background is briefly discussed to set the scene for the specific technical problem for this research. This technical problem is elaborated upon and formally stated as the research problem from which a number of objectives are set. The research methodology used to meet these objectives is then discussed and finally an overview of the structure of this thesis is presented.

Research Contributions of Chapter

Each chapter in this thesis begins with a model which is a representation of the content of that chapter and its contribution to the overall structure of this thesis and research. The model shows key words and their relationship to other, with the keywords summarising the content of the chapter shown in bold. The relationships show the logical links between the keywords and hence the research areas. The model conforms to Logical Modelling Notation which seeks to capture the exact meaning of the definition within the limitations of natural language. Logical Modelling is introduced in greater detail later in this thesis and detailed in Appendix K. The sentences on which the logical model for this chapter is based are shown below with the keywords shown in bold.

*This **Research** has established a **Problem** and **Objectives** to develop an **Architecture** of an **Approach** for a **Decision Support System**. The Approach will match **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

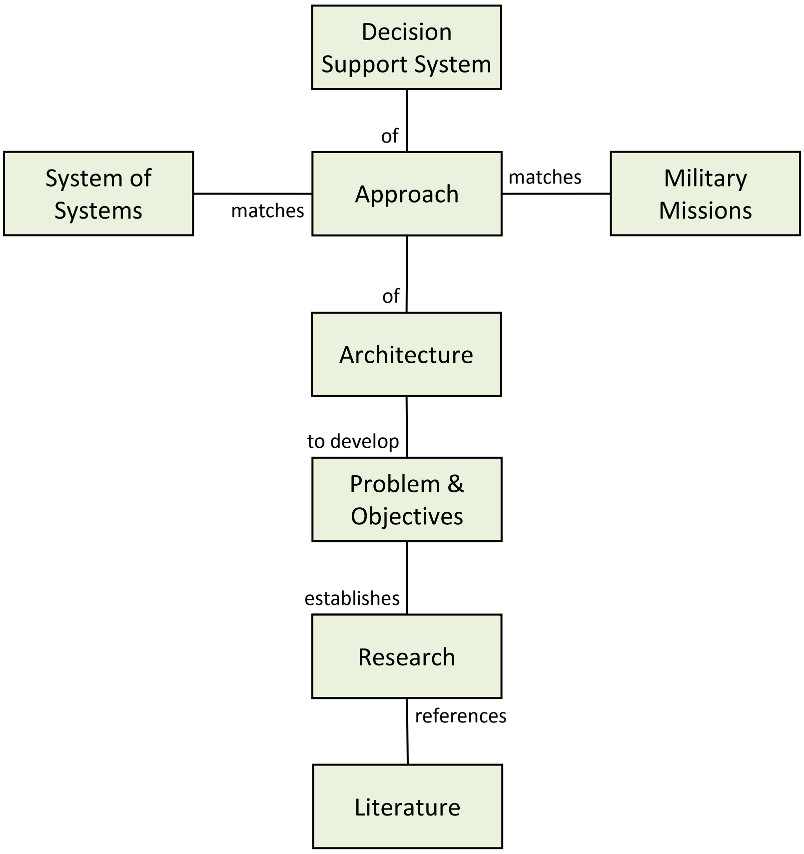


Figure 1 – Logical Model of Research Contributions of Chapter One

The keywords shown in Figure 1 cover the contents of this chapter and the lines between them indicate the relationships between the keywords. As the keywords shown in Figure 1 are new they are all highlighted. As this model is built on in subsequent chapters only the new keywords added to the model will be highlighted to show the contribution of each chapter.

1.1 Introduction

The history and key milestones of systems engineering have been well documented by the main organisations including the Institute of Electrical and Electronics Engineers (IEEE), International Council on Systems Engineering (INCOSE) and the United States Department of Defense (US DoD). In the literature the modern history of systems engineering has been well documented, for example Brill's 1950-1955 history (Brill 1998). To return to the start of modern systems engineering: according to Ackoff we have undergone a "Systems Revolution" as we have transitioned from the "machine age" to the "systems age" (Ackoff 1974, 2-4). Jackson and Keys (1984) summarise Ackoff's paper as follows:

[redacted]

(Jackson and Keys 1984, 476)

When Ackoff was concerned with systems which were *open* and cannot be understood using the methods of *reductionism*, he was proposing that a change in thinking was required from the analytical mindsets of the machine age, which derived an explanation of the whole of *something* through explanation of its constituent parts, to a synthetic mode of thought (or open systems viewpoint) that sought explanation of *something* by viewing it as part of a larger whole and explaining its role in terms of that larger whole. See Ackoff (1974, 3). The first part of Ackoff's statement encapsulates the reductionist mode of thought or the *closed* system viewpoint. Note that the *something* referred to in the previous sentences is not bound to be a physical object, but could be elements of any kind, e.g. concepts, ideas, people.

To return to Jackson and Keys summary of Ackoff consider a definition for machine:

[redacted]

(Oxford English Dictionary 1989)

A machine could be considered to be a physical device that transmits mass and energy to perform an activity. This machine viewpoint leads to a more common use of the terms *open* and *closed* in which these terms refer to attributes of the system in question. For example von Bertalanffy (1950, 23): "A system is closed if no material enters or leaves it; it is open if there is import and export and, therefore, change of the components". This is in contrast to Ackoff who considered that an *open* or *closed* system is a *viewpoint* on that system.

The central theme of this thesis can be stated by paraphrasing Ackoff (1974, 3): consider an open viewpoint that seeks to explain *a system* by viewing it as part of a larger *whole* and explaining its role in terms of that larger *whole*.

1.1.1 General Systems Theory

The "synthetic mode of thought, when applied to systems problems, is called the *systems approach*" (Ackoff 1974, 3). The systems approach integrates the reductionist and the synthetic method, encompassing both holism and reductionism. Note that there is a large body of literature that purports to apply the systems approach but there is a lack of other definitions as to what a *systems approach* actually is. For the purpose of this discussion Ackoff's definition will be used. The systems approach captures the way in which the pioneers of the general systems movement in the early fifties were operating. Among the peers of this group were Ludwig von Bertalanffy, a biologist, and Boulding, an economist, both of whom wrote on General Systems Theory, as introduced by Boulding:

[redacted]

(Boulding 1956, 197)

General Systems Theory recognised that problems were systemic in nature and could no longer be neatly partitioned into separate sciences. A significant role of General Systems Theory therefore was to "facilitate communication between disparate fields of interest, i.e., to provide a common language with which to discuss systemic problems" (Boulding 2004, 127).

General Systems Theory sought to provide a common language by combining modelling and communication. Modelling provides precision while communication provides comprehensibility, both of which are a basic need of design (a planning process), as succinctly stated by Brooks:

[redacted]

(Brooks 1995, 234)

The need for such a common language with the comprehensibility of communication (natural language) and the precision of modelling is due to the problem of planning and systems. With a focus established for systems we now seek a definition for this term.

1.1.2 Defining Systems

The word system comes from the Greek *sustema* meaning reunion, conjunction or assembly (Francois 1999). Two of many definitions from the literature are presented below:

[redacted]

(International Council on Systems Engineering 2007)

[redacted]

(Hitchins 2003)

The definitions shown above convey a sense of what systems are about but they are each incomplete. These definitions were used by Dickerson (C. Dickerson 2008) to derive through logical modelling a formal, logically consistent definition for systems. Logical modelling is utilised in the latter stages of this research in section 5.3, Logical Modelling; further details on logical modelling may be found in Appendix K, Logical Modelling . The formal definition proposed by Dickerson is shown below.

[redacted]

(C. Dickerson 2008, 3)

Note that whilst this provides a logically consistent definition for a system it does not necessarily tell us if an assemblage is an *open* or *closed* system as defined by Ackoff (1974, 3). The logical expectation is that any definition of systems should not tell us whether an assemblage is open or closed as that is a viewpoint on the system, not the system itself. The question of characterisation of systems has been raised in the literature, a cross section of which is shown in Ford et al (2009) but has yet to be universally agreed.

The consideration of whether a system is open or closed is not captured within the established definition of *system*. The open and closed aspects are an attribute of the viewpoint on a system, that is to say a system could be viewed as either open or closed. Hence a systems approach needs to be able to address the larger whole, using an open system viewpoint, while still maintaining a bridge to the closed system viewpoint to allow the system of interest to be viewed from both viewpoints. With *system* discussed we now consider combinations of systems to form *system of systems*.

1.1.3 System of Systems

Systems can be combined through interactions into a *system of systems* to achieve high level purposes unachievable by individual systems. Underlying the importance of the concept of system of systems (SoS) a dedicated IEEE academic forum has been established to explore this area of research (De Laurentis, et al. 2007). Further information on the history of SoS can be found in Gorod, Sauser and Boardman with their review of the modern history of SoS (Gorod, Sauser and Boardman 2008) which is built upon in Brill's work (1998). The Defense Acquisition University neatly describe the objective for, and characteristics of, a system of systems:

[redacted]

[redacted]

Chapter 4.2.6 - System of Systems Engineering(Defense Acquisition University 2006)

Jamshidi presents an overview of the six most commonly used definitions from the literature and references to many more (2005) but there is currently no universally accepted definition of system of systems (Sage and Biemer 2007, 6). A point of agreement among many systems and stakeholders is that "if the term system were correctly defined then it could be applied to itself to give a meaningful definition for system of systems" (Dickerson and Mavris 2009). Applying the definition of systems (C. Dickerson 2008, 3) to itself results in the following definition:

[redacted]

(Dickerson and Mavris 2009)

Just as Ackoff (1974, 3) noted that systems are inclusive in their scope and are not bound to physical objects, but elements of any kind, e.g. concepts, ideas, people, so it can be said of system of systems. The United States Department of Defense has explicitly identified the social aspects of SoS as one of the main challenges of realising SoS:

[redacted]

(Office of the Deputy Under Secretary of Defence for Acquisition and Technology, Systems and Software Engineering 2008, 7)

Therefore system of systems should not be considered only in terms of physical objects either and have found useful employment in other disciplines. For example, François presented a literature review of systemics and cybernetics (François 1999) in which the philosophical underpinnings of systems thinking is traced through various disciplines including psychology. The formulation of Gestalt psychology, the psychology of perception and forms, has systemic underpinnings as "perception must start by picking up static structures and dynamic interrelations between elements, i.e. is systemic" (François 1999, 206). The word *gestalt* in German literally means "shape" or "figure", but in English takes on another meaning of *wholeness* which is the basis for Gestalt psychology, that is the brain is holistic and the whole is different from the sum of the parts.

Gestalt psychology is of interest here because these concepts of psychology can reach into physical science. As Gorod et al noted, "Boulding imagined SoS as a *gestalt* in theoretical construction creating a *spectrum of theories* greater than its parts" (Gorod, Sauser and Boardman 2008, 486). Boulding also notes that such *gestalts* have "been of great value in directing research towards the gaps which they reveal" and gives the Periodic Table in chemistry as an example. He continues to note that SoS "might be of value in directing the attention of theorists towards gaps in theoretical models, and might even be of value in pointing towards methods of filling them" (Boulding 1956, 129).

To repeat the central theme of this thesis (paraphrasing Ackoff (1974, 3)): consider a synthetic mode of thought that seeks to explain *a system* by viewing it as part of a larger *whole* and explaining its role in terms of that larger *whole*. System of systems are more problematic to explain due to their scale and interactions/interdependencies of their constituent systems and the wider social and organisation issues. SoS present a series of individual problems related to the constituent systems which can be considered from a closed viewpoint. From an open viewpoint the SoS also presents social and organisational issues. Such interdependent problems combined with the social and organisational aspects have been identified in the literature and are discussed in the next subsection.

1.1.4 Wicked Problems

According to Ackoff, with the advent of the systems age problem solvers are increasingly faced not with separable problems, but rather with systems of interdependent problems, or *messes* (1974, 5). This correspondingly necessitates a holistic response to the problem:

[redacted]

(Ackoff 1974, 5)

The concept of the solution to a complex problem being more than the sum of its parts is a familiar systems concept. The critical point is that the aim of this exercise is one of satisficing (Simon 1979, 13) rather than optimisation to the main goal. Hence, whilst solutions to messes can be sought from a holistic perspective using models, the utility of such exercises is at the communicative level (as per Boulding) rather than the absolute level. Ackoff states that the "attempt to deal with a system of problems as a system – synthetically, as a whole – is an essential property of *planning* in contrast to problem solving" (Ackoff 1974, 5). To paraphrase Ackoff: an essential property of dealing with a system of problems from an open systems viewpoint is planning.

Prior to Ackoff's paper Rittel and Webber had noted that "planning problems are inherently wicked" (Rittel and Webber 1973, 160). They elaborate:

[redacted]

(Rittel and Webber 1973, 160)

Rittel and Webber suggest that planning problems, particularly societal problems, are inherently wicked. They identify engineering and science problems as being bounded such that the objective is clear and hence, due to this clarity, the solution can be identified. A systems engineer or systems architect face systems of problems (Ackoff's *mess*) where the mission may not be clear and, in turn, uncertainty may exist as to whether the problems have been solved. In the context of this research planning is the architecting of an approach to solve the system of problems from an open system viewpoint. Note that it is the attributes of the whole which imparts wickedness. Planning defines the approach to solve the system of problems <u>before</u> starting to solve the individual problems which necessitates a closed system viewpoint. The implication of this is that any system of problems dealt with from an open systems viewpoint will likely be wicked as defined by Rittel and Webber (1973), the distinguishing properties of which are summarised in Appendix A. If these *are* wicked problems then the aim is not to optimise but rather to find the good as opposed to bad solutions (as according to the third of Rittel and Webber's (1973) characteristics of wicked problems). Such wicked problems do not have an exhaustively describable set of potential solutions as according to the sixth of Rittel and Webber's (1973) characteristics of wicked problems.

By combining modelling and communication a bridge can be formed between the open system viewpoint of the architect and the closed system viewpoint of the engineer, as General Systems Theory set out to achieve. This would allow the wicked planning problem of the whole to be addressed using an open system viewpoint while still maintaining a bridge to the closed system viewpoint required to solve individual problems. Overall this approach would be an application of a "synthetic mode of thought" to systems problems (Ackoff 1974, 3) and hence would be a *systems approach*. With the rationale established for the need for a systems approach to deal with such systems of problems an initial statement of the research problem can be made, as discussed next.

1.1.5 Initial Statement of Research Problem

From the preceding discussion an overview of the academic area of interest has been given and an initial research theme identified and developed. This theme is the basis for the initial statement of the research problem as stated below:

Is there a decision support system that can analyse a system of systems as part of a larger whole from both open and closed viewpoints?

Initial Statement of Research Problem

This problem is the initial statement as it is at the conceptual level. To enable meaningful research to be conducted it needs to be restated at a more practical level where research can be conducted and later, utilising the understanding gained from this application, the conceptual findings can be derived. The practical application that this initial research problem will address is the subject of the next subsection.

1.2 A Research Application

This research was funded by the Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC), established by the UK Ministry of Defence (UK MoD). There are two distinct phases of this research, the original approach using traditional systems engineering approaches and methodologies, and an advanced approach using next generation systems engineering techniques. The advanced methods explored and developed in this research have been demonstrated useful but it is beyond the scope of this thesis to compare the improvements of these advanced methods over the original ones. The Combat Search and Rescue mission, which will be introduced in section 2.5, was chosen in line with the SEAS DTC as the instantiation of the practical application of this research.

The military has always been subject to change in the way it conducts war as the underlying technology and capability of the systems at its disposal have evolved: "War is a product of its age. The tools and tactics of how we fight have always evolved along with technology" (Alberts, Garstka and Stein 1999, 1). Recent military advances such as the tenets of Network Enabled Capability (NEC), or Network Centric Warfare (NCW) (Alberts, Garstka and Stein 1999), have highlighted the untapped potential capability of existing and proposed systems through more effective interoperability between systems to form system of systems (SoS). This potential capability can be realised through synergy between independent systems to enable the desired overall system performance (Abel and Sukkarieh 2006). This concept is significant enough that in 2006, the Deputy Under Secretary of Defense for Acquisition and Technology charged the Systems and Software Engineering Directorate to develop a guide for SoS, "recognising the value of systems engineering as a key enabler of successful systems acquisition and the growing importance of systems interdependencies in the achievement of war fighter capability" (Office of the Deputy Under Secretary of Defence for Acquisition and Technology, Systems and Software Engineering 2008, iii). By working within an SoS rather than as an independent system the advantages of a collaborative information environment can be realised and the individual capability of a system leveraged within the architecture of the SoS.

Two types of military SoS have been identified, *dedicated SoS* which have been consciously engineered to fulfil a need (e.g. an air traffic control system) and *virtual SoS* which are SoS created to support a specific military operation (Cook 2001, 3). Specifically, virtual SoS are characterised where "an SoS is mostly constructed at short notice based on available equipment or capabilities to meet an immediate mission requirement" (Chen and Clothier 2003, 173). Whilst conceptually the benefits of forming SoS at short notice are appealing, the practicalities involved with achieving this are difficult to realise. As Chen and Clothier note: "another feature of a virtual SoS is that it is regularly dismantled following operational deployment" (2003, 173). Essentially then a virtual SoS could be considered to be a temporal SoS created to fulfil a specific need (a mission in a military context) and encapsulates the entire SoS lifecycle from cradle to grave in a compressed timescale as determined by the specific need it addresses. A *dedicated SoS* can be thought of as a closed viewpoint for which a reductionist approach can be taken in line with von Bertalanffy (1950). A *virtual SoS* however, as a temporal creation requires an open viewpoint as espoused by Ackoff (1974).

Given the apparently small time scales involved the main barrier to forming such a virtual SoS is a fast route to establishing "good" SoS architectures that enable the collaborative capabilities of the constituent systems required by the need that the SoS fulfils. This point recognises that not all the capabilities of systems may be collaborative, either because they are unsuitable for distributed implementation or because of the so called *stovepiped* nature of their realisation which directly prevents them from interoperating within a SoS. Hence, combining systems into a SoS is currently a difficult undertaking requiring considerable expertise and experience. There is an identified lack of process for SoS and whilst there has been some starts made towards this end, e.g. (Sage and Biemer 2007, 5), one has yet to be formalised. Considering the temporal nature of a virtual SoS a methodology to elicit the requisite SoS architecture to enable the military to conduct missions would seem to be a fundamental gap in our ability to support SoS decision makers. Hence, this is the focus of this research. The problem is compounded by the generally held consensus that those in positions of power with regard to a system do not really care about how the system is constructed; they are rather more concerned with its performance characteristics:

[redacted]

(Boardman and Sauser 2006, 118)

Recognising that in an SoS characteristics such as survivability would be a product of the whole of the SoS presents a conflict between the need for careful formation of the SoS to achieve the requisite capability to conduct a particular mission (which will be dependent on the component systems) and the lead individual's concern with the emergent characteristics of the SoS. These two opposing views, top down and bottom up, must be reconciled through a middle ground. Later discussions will cover the issue of decision making and recently emerging military changes such as the proliferation of decision makers within the battlespace and the apparent shift from traditional hierarchical decision making processes towards faster, more fluid models. This research will aim to support these two transitions through a fast, usable and lightweight decision support tool to support, rather than dictate to, the decision maker. A discussion on the concepts of evaluating the SoS follows next.

1.2.1 Evaluating System of Systems

With the concepts of systems and SoS introduced the question becomes one of purpose for this research. A military conflict, at the lowest tactical level, could be thought of as two competing virtual SoS pitted against each other within an environment; in such a situation which SoS will prevail? Utilising techniques such as functional analysis an appreciation can be gained of the strengths and weaknesses of the component systems within each SoS. This will give an indication of the capability of each, but remember that for a SoS such a reductionist approach will not suffice, rather a synthetic approach is required. This requirement is compounded as the most influential aspects of a system are not necessarily the purely functional or even quantifiable. They could be the somewhat more abstract characteristics, the qualitative non-functional traits exhibited by the system.

The need for such non-functional characteristics has been alluded to through the use of the 'ilities' in requirements engineering. The 'ilities' refer to the set of terms ending in 'ility' which can be used to characterise systems, e.g. reliability, manoeuvrability, adaptability. Various authors have identified individual 'ilities' (Rhodes 2006) and their importance for SoS Engineering (Saunders, et al. 2005). However, there is currently no common, consistent and formally defined set of non-functional characteristics, including the 'ilities', which can be used to characterise SoS in commonly understood terms. This research contributes to a SEAS DTC project entitled "Impact of Different Cultural Attribute Sets on Semi-Autonomous and Autonomous System Decision Structures and Interfaces". One thread of work within the project focuses on the impact of 'soft factors' relating to cultural values on communicating and implementing decisions and is described in more detail in Siemieniuch and Sinclair (2006). The implications of this research are that cultural values can impact the ability of systems to conduct missions. When these systems are formed into a SoS there is a cumulative effect that can impact the ability of the SoS to conduct the mission (Siemieniuch and Meese 2006). A later paper based on a continuation of this work states that "there is an increasing recognition of the potentially deleterious effects of incompatible individual and organizational cultures on complex systems and organizations" (Hodgson and Siemieniuch 2008). This supports the notion that non-functional characteristics of a SoS may be as important as its functional characteristics in determining its suitability to conduct some task.

In the military domain, which this research is focused on, tasks are normally expressed as missions. Military organisations typically have many systems available which can be combined and configured to form multiple SoS, any of which could be used to conduct a mission. Which of these SoS from both functional and non-functional viewpoints are suitable to conduct the particular mission? This choice is routinely faced by military decision makers. How they currently make this decision is discussed in detail in the next chapter (in section 2.3 Review of Military Decision Making), but for now it is suffice to say that they lack support in terms of being able to profile the SoS alternatives using a transferable, commonly understood set of system characteristics which can be applied to both SoS and the mission to allow easy comparison and to highlight capability gaps between "what is needed" and "what we've got." Current military decision support methodologies rely on the experience of the decision maker to understand the systems/SoS in specific scenarios. At the SoS level the variables of interest in this decision are the selection of the systems comprising the SoS and the configuration of their organisation and interconnections. SoS can be represented using system architectures to capture the pertinent information concerning these variables, which are elaborated upon in the next section.

1.2.2 Systems Architecting

It is important to differentiate between systems architecture and systems architecting. The systems architecture that this research will consider is in the military context and hence will address military missions as discussed in the next subsection. For the purpose of this section a systems architecture can be thought of in the same way as a cartographer's map, which is not the terrain but rather a useful abstraction for the purpose of navigation. A system's architecture deals with the interdependence of the chosen components and the overall functionality of the system, at a level of abstraction that is useful to the architect. As Maier and Rechtin (2000) states, "architecting deals largely with unmeasureables using non-quantitative tools and guidelines based on practical lessons learned; that is, architecting is an inductive process. At a more detailed level, engineering is concerned with quantifiable costs, architecting with qualitative worth" (Maier and Rechtin 2000). Qualitative worth implies that the level of detail required of individual systems is relative to the impact that they will have on the overall SoS in achieving the objectives and purpose of that SoS (and therefore ensuring client satisfaction). From Maier and Rechtin (2000):

[redacted]

(Maier and Rechtin 2000).

But what characteristics of a system are important with respect to the function, cost and timeliness required for success? How should these characteristics be measured? As discussed, at an architectural level it is the qualitative worth of a system that is important, as opposed to the precise details.

In this research the objective is not to establish a single-point mathematically based optimising methodology, but one rather more based on satisficing. The complexity inherent in system designs and the high level of uncertainty associated with the systems which comprise the SoS and the environment usually requires robust optimal solutions to be found in most cases, rather than sensitive, single-point solutions. This is founded in the principal that, within the constraints under which the SoS operates, there is a need to make a "good" decision among alternatives.

1.2.3 Systems Architectures And Military Missions

A military mission, simplistically, involves achieving some purpose and objectives in an uncertain environment where a friendly force and an enemy force exist in competition with one another. In simplistic terms, mission success for a friendly force could be thought of as requiring it to meet and exceed the capabilities of the enemy force and, in so doing, meet the mission's objectives. This "positioning for capability superiority" is similar in concept to the positioning for information superiority model proposed by Alberts, Garstka, and Stein (1999), as shown in Figure 2.

[redacted]

Figure 2 – Positioning for Information Superiority(Alberts, Garstka and Stein 1999)

This model deals with gaining information superiority over an adversary, with three key metrics proposed: relevancy of information, accuracy of information and timeliness (of information). In simplistic terms, by outperforming the enemy for each of these metrics, information superiority is achieved. However, in the context of military operations, the mission has to be assessed in terms of numerous criteria. Similarly, the SoS assembled to carry out the mission has to be judged in terms of the capability which it needs for superiority over the enemy forces. This capability could be thought of as an expression of the combined functional and non-functional system characteristics that comprise the SoS, as discussed earlier. The concept of capability superiority fits with Rechtin (1999) who defines such competition in architectural terms as, "an attempt by one system or organisation to equal or surpass others to gain something of value" (Rechtin 1999). In terms of this research the "something of value" would be defined in the mission's objectives. This positioning for capability superiority is shown in Figure 3, where the SoS's capability signature envelops the mission demand's capability signature, hence achieving capability superiority. More rigorously, the comparison requires an investigation of a feasible trade space. This multi-dimensional space is difficult to represent in two dimensions, but the spider diagram format of Figure 3 helps to interpret the comparison.

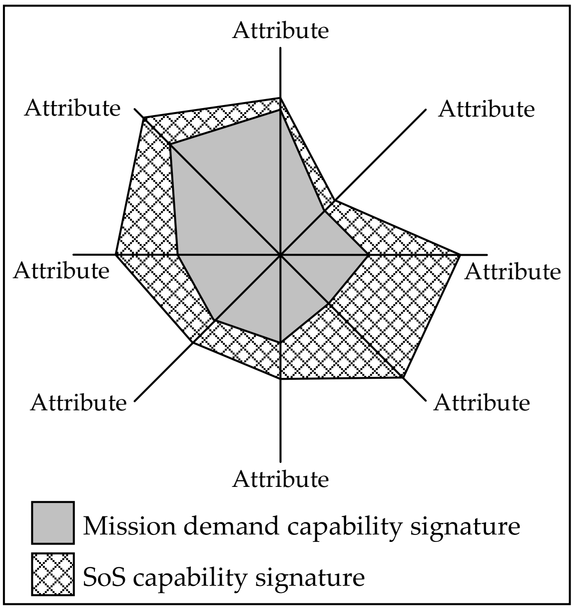


Figure 3 – Positioning for Capability Superiority

The idea that mission success could be enabled (as opposed to guaranteed) by considering competitive architectures, in terms of qualitative capabilities, is also lent credence by Rechtin. On the subject of competitive systems Rechtin (1999) states, "Like economies and the art of war, it is primarily about relative levels between the competitors' capabilities rather than about their absolute values, sizable and important as the latter may be" (Rechtin 1999). This, in the context of military missions, would indicate that a higher level of abstraction will suffice for comparing competing SoS, or comparing SoS in the application of "what we need" against "what we've got". To achieve the view of capability superiority as illustrated in Figure 3 a commonly understood set of functional and non-functional system characteristics are required as well as a framework to allow the effective assimilation of SoS capability signatures from the system architecture. This research considers what these system characteristics might be, how they could be measured and how they could be used to support a decision maker to decide what architecture to use for a particular mission through the utilisation and development of systems ideas to this application. With the concept of capability superiority introduced we now pursue a definition for capability which is the focus of the next subsection.

1.2.4 Capability

Capability has been defined by the UK Ministry of Defence as:

[redacted]

(Ministry of Defence 2009)

Capability is delivered by force elements combined into packages (system of systems). This research considers how to rapidly assess these packages in terms of the required capability. While the UK Ministry of Defence considers the general problem of working within a coalition this research will be bounded to consider the particular problem of indigenous force elements. This is with the expectation that the developed approach may be extended at a later stage to include coalition forces. The US DoD's Joint Concept Development and Revision Plan contains another definition for *capability* which implies methods for assessing packages:

[redacted]

(Department of Defense 2004, 15)

Capability moves away from a *means* based perspective (e.g. platforms and systems) to a combined *ways and means* perspective. This perspective does not consider the systems but rather the overall capability required to perform a mission. Hence, to assess these packages both the *ways* and *means* must be assessed in some way. To a degree this alleviates the problem of Maslow's Maxim (also known as the silver bullet):

[redacted]

Abraham Maslow (Thornton-Wells, Moore and Haines 2004)

In military terms the hammer and nail of Maslow's Maxim are analogous to platforms and their effects. By moving to a capability based perspective of problems the solution can be expressed in terms of capability which a platform may or may not be able to fulfil. The platform is no longer the driver of the solution (and can no longer be proposed as a *silver bullet* to those problems). Figure 9 shows the components of capability as according to the UK Ministry of Defence.

[redacted]

Figure 4 – The Components of Capability (Ministry of Defence 2009)

Note in Figure 4 the multiple viewpoints on capability of which the joint capability packages, or system of systems, is only one. To return to Maslow's Maxim the problem here is captured by the *Threat* and *Physical Environment* and the answer by the *Joint Capability Packages* and *Coalition Contribution*. An appropriate capability can therefore be realised by matching the mission demand (incorporating the *Physical Environment* and *Threat*) and the SoS (incorporating the *Joint Capability Packages* and *Coalition Contribution*) as proposed in the previous section.

1.3 Statement of Research Problem

Previously an initial research problem had been identified from the academic literature:

Is there a decision support system that can analyse a system of systems as part of a larger whole from both open and closed viewpoints?

Initial Statement of Research Problem

When this research talks of a system capable of supporting both open and closed viewpoints it shall be referred to as a *holistic* system. This conceptual problem has been considered in a military application where a need has been identified to be able to match alternative SoS architectures to particular mission requirements in order to support military decision makers. This is based on the lack of processes to support SoS formation (Sage and Biemer 2007) which has been identified as a critical shortcoming by the author due to the temporal aspects of virtual SoS formation for particular missions, as described by Cook (2001) and Chen and Clothier (2003). The research problem is stated below:

Is there a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission?

Statement of Research Problem

A systems approach will be taken to create an architecture for a decision support system that can analyse a system of systems as part of a larger whole from both open and closed viewpoints. Note that this is not the solution (as this research addresses wicked problems and there are no solutions) but rather an architecture of a systems approach to the problem. With the research problem established a research methodology was defined which is outlined in the next section.

1.4 Research Methodology

[redacted]

John Stuart Mill

The epistemology, or theory of knowledge, most heavily drawn upon in this research is critical rationalism, as advanced by Karl Popper. Popper advocated empirical falsification over the classical observationalist/inductivist account of scientific method. Popper's philosophy of critical rationalism was "the first non justificational philosophy of criticism in the history of philosophy" (Bartley 1964, 23). The core concept is captured by Taleb (2007):

[redacted]

[redacted]

(Taleb 2007, 126)

This research does not seek a solution, rather it seeks a decision support system that can view a system as part of a whole, a system of systems, from both open and closed viewpoints. Hence it cannot be proved – only stated such that its use can be demonstrated and, most importantly, can be falsified. A solution cannot be pursued as the problem is wicked and hence unique each time so an empirical proof would only hold for that particular instantiation of the problem, not the problem as a whole. So whilst the decision support system cannot be proved right it will at least be falsifiable.

1.5 Research Method

The research method followed by this research is in line with traditional methods of scientific research to the extent that they can be applied to open and wicked problems. The major adjustments to address the wicked nature of the problem this research addresses are in stages four and seven.

1. Formation of the research topic - *open and closed viewpoints, systems and system of systems*

   The background for this research is discussed in subsections 1.1 and 1.2.
2. Hypothesis – *towards a holistic decision support system*

   The hypothesis of this research is that there is a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission, as stated as the research problem in subsection 1.3. The decision support system this research seeks to develop is holistic due to the linked open and closed viewpoints it will consider. The proof of existence of such a decision support system will be through the existence of an architectural approach that informs the solution.
3. Conceptual definitions – *military missions, functional and non functional characteristics*

   Defined in Chapter 2 and Chapter 3.

4. Architectural approach – *the parametric approach / model driven approach*

   Defined initially in Chapter 3 and Chapter 4 and revisited in Chapter 6.

5. Gathering of data – *historical case studies*

   Started in subsection 2.5 and extended in 4.3. Narratives for the historical case studies may be found in Appendix I.1 and J.1.

6. Analysis of data – *historical case studies conducted with developed decision support tool*

   First set of case studies completed in subsection 4.3 and then reanalysed in Chapter 7.

7. Test and revision of hypothesis - *testing falsification from an open viewpoint*

   Testing is performed initially in Chapter 4 and with the second developed decision support tool in Chapter 7. The hypothesis is revised with the findings from this first set of case studies in Chapter 5. The testing of the hypothesis is conducted in line with critical rationalism and hypothesis is capable of falsification. This is what distinguishes this research method as traditional reductionist evaluation methods are inappropriate for this research.

8. Conclusions – *findings on a wicked problem*

   The conclusions to this research are reported in Chapter 8 along with future research opportunities.

1.6 Objectives

Seven objectives were set to answer the research problem stated previously in subsection 1.3 using the research method outlined in 1.5. Each objective is listed in turn as a separate sub heading and briefly described.

1.6.1 Objective 1: Review General Decision Making

This research aims to develop an approach for a decision support system. To undertake this an appreciation is required of how people make decisions and a review of general decision making from the literature is required.

1.6.2 Objective 2: Review Military Decision Making

The context of this research is military orientated. The author recognises that military decision makers may make decisions differently from their civilian counterparts. This review will allow military decision making literature to be assessed and compared to general decision making theory.

1.6.3 Objective 3: Identify a Decision Making Model to Support

To develop an architecture of an approach for a decision support system a specific decision making model must be identified to support. This objective is to identify, after reviewing the general and military decision making literature which model is most appropriate to support, with justification of course.

1.6.4 Objective 4: Identify a Practical Application for this Research

A specific practical application is required for this research, firstly to help guide the development of the approach and secondly to provide a specific context for the case studies. This objective is to identify an appropriate practical application for these purposes.

1.6.5 Objective 5: Develop an Architecture of an Approach for a Decision Support System

With a decision making model identified to support and a practical application established this objective is to develop an architecture of an approach for a decision support system.

1.6.6 Objective 6: Implement the Developed Approach

This research considers a wicked problem in so much that there is no particular right answer. The architecture of an approach for a decision support system established by the previous objective is not practically usable until it is implemented in some manner. This objective is to implement the developed approach.

1.6.7 Objective 7: Conduct a Set of Case Studies with the Implemented Approach

The final objective of this research is to conduct a set of case studies with the implemented approach to assess its validity using the chosen research methodology of critical rationalism.

This thesis will address the success of the research in terms of achieving these objectives in Chapter 8, section 8.2.

1.7 Contribution of this Research

This research will ultimately demonstrate the novel application of a model driven approach to decision support. Specifically, this will allow the decision maker to see how their proposed solution (expressed as a system of systems) matches the mission's requirements. This is initially achieved through a parametric approach as discussed in Chapter 3 that identifies three sets of characteristics (functional, non-functional and physical environment) that can be applied to Systems of Systems and mission requirements to allow them to be matched. The resultant findings from a set of case studies (shown in Chapter 4) conclude that whilst the sets of characteristics are useful the parametric approach is flawed. An alternative to the parametric approach is a model driven approach, as introduced in Chapter 5. A model driven approach for the functional set of characteristics is developed in Chapter 6 and the set of case studies originally conducted with the parametric approach are repeated with the model driven approach in Chapter 7. It is this novel application of a model driven approach to decision support as developed in Chapter 6 and tested in Chapter 7 that is the main contribution of this research. Further contributions that this research makes are discussed in the conclusions in Chapter 8.

1.8 Structure of Thesis

The Research Problem developed in this Chapter is expanded upon in Chapter 2 and a practical application identified. An architecture of an approach for a decision support system is developed in Chapter 3. Chapter 4 documents the implementation of the developed architecture and two initial case studies are conducted using the developed software tool. The findings from the first set of case studies indicated a different approach was required from the initial architecture developed. Chapter 5 addresses these findings by developing an architecture of a model driven approach for a decision support system. The implementation of the architecture of a model driven approach is presented in Chapter 6. This second implementation is evaluated by repeating the two case studies, which is covered in Chapter 7. Chapter 8 reviews the research reported in this thesis, revisits the original research question and objectives, outlines future research opportunities and finally draws to a close with a set of conclusions. A Glossary of the acronyms used are provided in Chapter 9. The References cited in this Thesis are shown in Chapter 10. Finally, the Appendices are contained in Chapter 11 which starts with a separate table of contents for them.

1.9 Chapter Summary

This Chapter started with a discussion of the nature of systems problems and the need for an open viewpoint to explain a system by viewing it as part of a larger whole and explaining its role in terms of that larger whole. This was linked to General Systems Theory which sought to provide a common language to discuss systemic problems by combining modelling and communication; modelling to provide precision while communication provided comprehensibility. An overview of the definition of *systems* and *system of systems* was covered and led to a discussion of *wicked problems*. By combining modelling and communication a bridge can be formed between the open system viewpoint of the architect and the closed system viewpoint of the engineer, as General Systems Theory set out to achieve. This would allow the wicked planning problem of the whole to be addressed using an open system viewpoint while still maintaining a bridge to the closed system viewpoint required to solve individual problems. Overall this approach would be an application of a "synthetic mode of thought" to systems problems (Ackoff 1974, 3) and hence would be a *systems approach*. A practical application for the research was discussed and a need identified for this approach within the military domain to support decision makers. The research problem was stated with this practical application focus:

Is there a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission?

Statement of Research Problem

Chapter 2

Problem Definition and Practical Application

Outline of Chapter

This chapter considers decision making in a general context, specifically the types of decision making situations that humans undertake and the viewpoints on modelling decision making. The main decision making under risk theories are surveyed and a general need to support human decision making behaviour justified. With the particular type of decision making that this research is interested in established and the type of model required to build a DSS around identified we then consider military decision making to understand how military decision makers currently make decisions. Emerging decision making models for the military are then considered to find one that accurately reflects how military decision makers are actually making decisions that this research can support.

Research Contributions of Chapter

As introduced at the start of Chapter 1 the sentences describing this chapter are shown below with the keywords highlighted which are shown in turn in a logical model on the next page.

From the previous chapter(s):

This ***Research*** *has established a* ***Problem*** *and* ***Objectives*** *to develop an* ***Architecture*** *of an* ***Approach*** *for a* ***Decision Support System****. The Approach matches* ***System of Systems*** *to* ***Military Missions****. This Research references a* ***Literature Review****.*

This chapter introduces:

The Decision Support System supports the ***Decision Maker*** *through the Recognition Primed Decision Making model. The* ***Application*** *for this research is the* ***Combat Search and Rescue*** *military mission.*

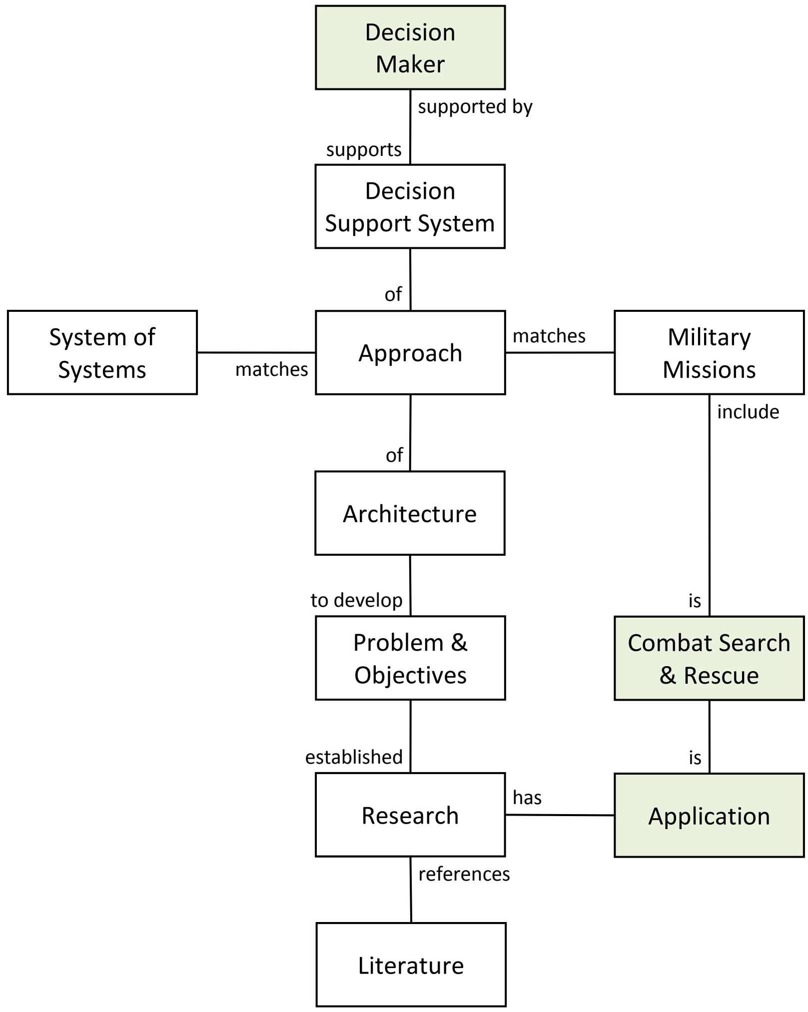


Figure 5 – Logical Model of Research Contributions of Chapter Two

The new keywords introduced in this chapter are highlighted in Figure 5. Hence, the keywords not highlighted in Figure 5 are from the model established in Chapter 1.

2.1 Background

This chapter considers decision making in order to identify a model of military decision making that the decision support tool this research aspires to develop can support, as outlined in Chapter 1. By understanding how military decision makers actually make decisions the decision support tool will be able to better match how they work and, in so doing, will hopefully be more usable by them.

Decision making is a considerable field of study and this chapter starts in subsection 2.2 by reviewing the general decision making literature to build up a picture of how humans make decisions and to start to identify any need for support for human decision making activities. The literature review presented was focused through reference to a more extensive review conducted by Hubbard (2010). The particular case of decision making in the military is considered in section 2.3, including a review of the major decision making paradigms. Combining the findings from the general decision making review with the particular case of military decision making subsection 2.4 considers future trends in military decision making models and settles on the model to be used in this research. The chapter finished with subsection 2.5 which outlines the particular military mission to which the decision support system will be applied.

2.2 Review of General Decision Making

Most decisions are gambles. They are choices between alternatives, the information about which are incomplete. The decision maker must therefore choose between the alternatives, the choice has not been made for them. The literature distinguishes between two particular types of decision making situations, those based on *risk* and those based on *uncertainty*. *Risk* is defined as "decision situations in which the probabilities of an outcome are objective or given, such as betting on a flip of a fair coin, a roll of a balanced die, or a spin of a roulette wheel" (Wu, Zhang and Gonzalez 2004, 399-400). Risk can be thought of as a closed viewpoint on a situation where reductionist methods can be applied to determine the probabilities of outcomes. *Uncertainty* is defined as situations in which the outcomes are subjective and the decision maker must estimate or infer the likelihood of an outcome occurring (Wu, Zhang and Gonzalez 2004, 399-400). Uncertainty can be thought of as an open viewpoint on a situation where the outcomes must be subjectively evaluated using synthetic methods. Most decisions involve risk, a comprehensive history of which can be found in Bernstein (1996). However, the decisions of interest to this research clearly involve uncertainty rather than risk; the nature of war does not lend itself well to quantitative risk assessment. The academic research on decision making considers three broad questions (Wu, Zhang and Gonzalez 2004, 399), which are:

- How *should* decision makers behave when faced with risk/uncertainty? (Normative)
- How *do* decision makers behave when faced with risk/uncertainty? (Descriptive)
- How can decision makers be *made* to act more normatively? (Prescriptive)

The research problem stated previously in subsection 1.3 refers to supporting military decision making. Supporting military decision making is not normative or prescriptive, hence these are out of this research's scope. To develop a decision support system requires a model of *how* decision makers behave – a descriptive model – to construct the tool around. This recognises that decision makers in the military can already make the decisions that this research considers, so the aim is to support rather than prescribe to them.

So how are decisions made in general? To start to answer this requires a move into the literature of economists. While we are primarily interested in decisions of uncertainty there is considerably more literature available on risk. Risk can be considered to start with as the argument has been made in the literature that "our understanding of the simpler situation of risk readily extends to the more realistic case of uncertainty" (Wu, Zhang and Gonzalez 2004, 400).

In economics the standard theory of individual choice is the *Expected Utility Theory* (Starmer 2000, 332). It has been described as "the major paradigm in decision making since the Second World War" (Schoemaker 1982, 529). Bernoulli proposed expected utility as a resolution to the St. Petersburg Paradox in his 18th century paper *Specimen Theoriae Novae de Mensura Sortis* (translated more recently by Dr Louise Sommer (Bernoulli 1954)). The St. Petersburg Paradox considered a risk based decision whereby someone pays a fixed price to play a game where the winnings depends on the number of tosses of a fair coin it took it took to get heads. The amount they received was 2^n, where n is the number of tosses of the coin to get heads and the probability of winning is $(½)^n$. The curiosity of the paradox was that whilst it is a game of infinite expected monetary value the results found that people were only willing to pay a small monetary value to play it. This was an irrational decision making behaviour. Bernoulli's theory was that people maximise expected *utility* rather than expected monetary value and presented a descriptive model to show this. As a descriptive model it had no practical value until von Neumann and Morgenstern "showed that the expected utility hypothesis could be derived from a set of apparently appealing axioms on preference" (Starmer 2000, 334). These axioms are detailed in Schoemaker (1982, 531-532). However, these axioms are normative and prescriptive, as opposed to descriptive.

Expected Utility Theory is an example of, and arguably the most prevalent of the so called *conventional* decision making theories. These theories "model choice as preference maximisation and assumes that agents behave *as if* optimizing some underlying preference function. The *as if* is significant here: the conventional approach, interpreted descriptively, seeks to predict which choices are made and typically, there is no presupposition that the model corresponds with any of the mental activities actually involved in making choices." (Starmer 2000, 349). While these conventional models of decision making dominate economic theory, psychology is more focused on modelling the process of making a decision. The most noted of these non-conventional theories is *Prospect Theory* which was developed by Kahneman and Tversky (1979) in response to their critique of Expected Utility Theory. There are two main features of Prospect Theory of interest to this research: the first is the *isolation effect* which is where the decision maker discards components that the alternatives share. This editing process is inconsistent between decision makers as a pair of prospects can be partitioned into shared and different components in multiple ways. The recognition that people can interpret the same situation in multiple ways is a theme which will be expanded upon later in this thesis when we consider Transformational Grammar (section 5.2). The second and more important feature of Prospect Theory is the consideration of outcomes as an interpretation of gains and losses relative to a reference point. This reference point could be taken to be the status quo, or one's current assets. As Kahneman and Tversky (1979) note: "Although this is probably true for most choice problems, there are situations in which gains and losses are coded relative to an expectation or aspiration level that differs from the status quo. For example, an unexpected tax withdrawal from a monthly pay check is experienced as a loss, not as a reduced gain." Such situations are illustrated in Figure 6 where $\chi = 0$ is the status quo, or reference point. Notice that the gains curve is concave and the losses curve is convex and steeper. Tversky and Kahneman (1991, 1047-1048) characterised these features as general behaviour of decision makers using the terms *Loss Aversion* and *Diminishing Sensitivity*. These features are characterisations of irrational decision making behaviour. Diminishing Sensitivity, which is the decrease in the marginal value from the reference point, helps describe the St Petersburg Paradox.

Figure 6 – The Valuation of Outcomes in Prospect Theory (Starmer 2000, 351)

The decision biases revealed by Prospect Theory are expanded upon in various other psychology literature surveyed on decision making. For the amusement of the reader two thought experiments are given in Appendix C which help illustrate these limitations. Various "cures" to these human decision making biases/limitations have been proposed and developed, including decision making methodologies to help us make decisions, decision support systems to monitor and guide human decision making, and "expert systems" that remove humans from the decision making process and, the theory goes, in doing so our human fallacies.

This research considers decision making under uncertainty and seeks a descriptive model of military decision makers for which a decision support system can be developed. Two main decision making theories from economics are the *Expected Utility Theory* which states that people maximise utility as opposed to reward and *Prospect Theory* which characterises the general behaviour of decision makers with the terms Loss Aversion and Diminishing Sensitivity. Both of these theories indicate that humans are innately irrational decision makers liable to be subject to a number of decision making fallacies. Note that this does not specify how the decision makers actually make the decision, rather it characterises the likely features of the decision made. A decision support system would be useful to help negate these decision biases. Aware of these limitations of decision making we now consider how decisions are made within the military and consider how best to support them.

2.3 Review of Military Decision Making

This subsection considers how decision makers in the military make decisions. The subsection starts with a consideration of general military decision making before looking at decision making in military operations in 2.3.1. The predominant decision making process, the Military Decision Making Process is discussed in 2.3.2. Finally the Observe Orient Decide Act Loop is discussed in 2.3.3 and extended to the emerging tenants of Network Centric Warfare in 2.3.4. To begin, decision making in the military is described in the United States Army doctrine:

[redacted]

The Military Decision-Making Process (Department Of The Army 1997, 5-1)

Decision making is dependent on information, the nature and use of which in warfare has changed so significantly in the past decade it has had a transformational influence on the way in which modern warfare is conducted. The value of information for decision making is defined as "the difference between a decision maker's payoff in the absence of information relative to what can be obtained in its presence" (Banker and Kauffman 2004). In the military domain the prevalence of information enabled relatively recently through the tenets of Network Centric Warfare (NCW) or Network Enabled Capability (NEC) has impacted the way in which decision making is conducted:

[redacted]

Implications for Military Operations, Network Centric Warfare (Alberts, Garstka and Stein 1999)

Note the underlying message in the above quote is that more information has a direct correlation to better decision making (however that may be measured). The increase of information availability has also decreased the time that decisions must be made in for them to be effective. Information has always been a crucial commodity to posses in warfare, arguably more so today than ever before because of its availability. Sun Tzu in his thesis The Art Of War emphasises the need for information about both the enemy and your own forces:

[redacted]

Sun Tzu, The Art Of War (Heinl 1966, 320)

In modern warfare the role of information has been elevated to that of a decisive factor in victory, as proposed through the original tenets of Network Centric Warfare (NCW) (Alberts, Garstka and Hayes, et al. 2001, Alberts, Garstka and Stein 1999). It has been proposed by Alberts, Garstka and Stein (1999) that information has the dimensions of relevance, accuracy, and timeliness, as shown in Figure 7.

[redacted]

Figure 7 – Superior Information Position (Alberts, Garstka and Stein 1999)

Positioning within these dimensions to exceed a competitor is described as *information superiority*:

[redacted]

Information Age Organizations, Network Centric Warfare (Alberts, Garstka and Stein 1999)

Information on the battlefield, about the enemy, the environment and your own forces, has been a decisive factor in the outcome of the great military campaigns of history. This is still true today where the battlefield has become the battlespace which encompasses air, land, sea and space as well as enemy and friendly force locations:

[redacted]

U.S. DoD Dictionary of Military and Associated Terms (U.S. Department Of Defense 2006)

The battlespace is a strategic level concept which can be considered at a lower, operational or tactical level by appropriately bounding the area of interest. Considering the scope of the definition of battlespace in the quote above and the amount of potential information that it would encompass, bounding is necessary to prevent information overload. War is often surrounded by confusion and a lack of information, the so-called fog of war. The Prussian military philosopher Carl von Clausewitz wrote:

[redacted]

- Carl von Clausewitz, On War (Alberts, Garstka and Hayes, et al. 2001, 36)

Whilst advances in communications and human machine interfaces have increased the availability of information in the battlespace the sheer quantity can threaten to overwhelm the recipient. Filters can be employed to simplify and reduce the incoming data but at the risk of obscuring or removing useful information. What is useful information in this context? What is the value of such information?

To help answer this we step into the world of Information Theory which began with Shannon's seminal paper (1948) which considered the transmission of information across a noisy communications channel. Shannon's paper (1948) contained many revolutionary ideas, including the use of binary digits, or bits, to measure information and the idea of information entropy. It is not this paper that is of particular interest to us, but rather a paper written a little later on which linked Shannon's theories to gambling. Shannon's paper was founded in cryptography and it considered coded messages; but could the theory be applied to situations where no coding was used? This was considered by another employee at Bell Labs, where Shannon worked, called John Kelly, Jr. Kelly was apparently inspired by news reports of a con involving an American TV show called The $64,000 Question. The latest incarnation of this show is Who Wants To Be A Millionaire? and the format is largely the same with contestants answering questions, each question doubling their winnings. The show was broadcast live on the East Coast but its broadcast was delayed by three hours on the West Coast, for scheduling and time difference reasons presumably. The bookies on the West Coast allowed bets to be placed up to the shows airing. The con involved West Coast gamblers placing bets on the winner of the show after finding out who had won from East Coast viewers (Poundstone 2005, 66). Kelly considered how a gambler with access to such "inside information" may best utilise it for maximum financial gain and founded his work in Shannon's theories (Kelly 1956).

[redacted]

The Gambler With A Private Wire,
A New Interpretation Of Information Rate (Kelly 1956, 918)

The scenario Kelly considers is placing bets on racehorses and shows how a gambler can maximise their return but is applicable to any application where profit can be made from having information not generally available. In the context of gambling and the stock markets "insider information" has rather negative, if not criminal, connotations. This is not necessarily the case:

[redacted]

Entropy, Fortune's Formula (Poundstone 2005, 76)

To return to a military context, inside information can be thought of as *intelligence*, which is information gained about the enemy that gives us some sort of advantage over them. This could be thought of as an *external* or *environmental* edge, which is an advantage gained through knowledge outside of our system boundary. In the research we are focusing on, if a military system were able to model or estimate the qualities of a system more accurately against mission requirements would this constitute inside information? By knowing more accurately what you are deploying to achieve some objective, and knowing how well the deployed system matches up to the requirements of that objective then some sort of edge would be gained. This could be thought of as an *internal* or *systemic* edge. This consideration of gambling is not entirely at odds with military operations; in fact they are somewhat analogous as stated by Carl von Clausewitz.

[redacted]

Carl von Clausewitz, On War(Graham 2008)

If war is viewed as a game of chance, a gamble so to speak, it would seem wise to understand the elements of the game that could contribute to success or failure. At a lower level of war, the tactical level at which individual missions are conducted, these elements could be thought of as functional (what we need to be able to do to achieve the mission) and non functional (the intangibles required for mission success). Whilst the consideration of these elements would not guarantee success, they would help indicate if the odds were in our favour or not. If we are not functionally capable of conducting a mission we would be unlikely to succeed. But what information would we need about ourselves and the enemy to achieve this? This question of characterisation is a key thread of this research and will be discussed later in this thesis.

As discussed previously humans can be susceptible to certain fallacies, as introduced in the previous subsection 2.2 and elaborated upon in Appendix C. That is not of course to say that we should not make decisions, our decision making ability generally helps avoid catastrophe (Perrow 1984). However, in times of highly stressful, time limited decision making (as exemplified by military decision making) we are most vulnerable to our innate psychological weaknesses. The key is the limitations that intrude on our ability to process information, as Klein (1999) states:

[redacted]

Why Good People Make Poor Decisions (G. Klein 1999, 274)

In the military context of this research, where a bad initial decision can be very costly, a decision support system that helps overcome these fallacies through clear, concise information presentation would seem to be advantageous. At a time when the information available to the decision maker is ever increasing the DSS will also help alleviate so called "paralysis through analysis", where the quantity of information overwhelms the decision maker and prevents decisions being taken. Hence, such a DSS could help alleviate the effects of stressors as identified by Klein.

Decision making is changing in the military due to the increased availability of information which is leading to better, but more time constrained decision making. War is often surrounded by confusion and incomplete information leading to increased uncertainty. The value of timely and accurate information about the enemy (*intelligence*) provides an informational advantage to the decision maker. Military decision making is typically time constrained and highly stressful which limit human decision making and can negate any informational advantage. A decision support system which could alleviate these effects and maintain any informational advantage would be useful. The following subsection considers decision making within the military and considers the contribution that the decision support system this research aims to develop could make.

2.3.1 Decision Making in Military Operations

Helmuth Karl Bernhard Graf von Moltke, who became Chief of the Prussian Großer Generalstab, oversaw the formation of the Generalstab (general staff) of the Prussian Army. The Generalstab was mainly responsible for the rapid defeat of France in the Franco-Prussia War of 1870. This defeat proved that large-scale wars could be won by planning detailed requirements, rapidly devising capabilities to meet them and implementing them exactly (Paparone 2001). This concept has been maintained and extended to *decision superiority*, as espoused by the US Military in their concept for future operations, quoted below:

[redacted]

(Joint Chiefs of Staff 2000, 8)

Decision-making during military operations has traditionally been done only by humans, supported by a few notable automated decision-aid tools used for specialist applications such as mission planning, reactive defensive aids, carefree handling of aircraft. Within the force structure, it is increasingly likely that decisions will be made by lower ranks acting as autonomous units, or autonomous individuals, in order to enable shorter response times in the face of varying threats. This is partly the result of force transformation to achieve greater agility, enabled by concepts such as network centric warfare, and partly in response to the emerging asymmetric (Ancker III and Burke 2003, 20) and non-traditional warfare threats, both of which require fast response time and decentralised decision making. We will touch on these areas later on in this section.

In Gary Klein's book Sources Of Power (1999) he includes anecdotal reasons for how his research company won a research contract from the U.S. Army Research Institute for Behavioural and Social Sciences. He talked to some program administrators at the institute who explained how, "the U.S. Government had spent millions of dollars in the 1970s and early 1980s finding out how people make decisions, and the army has used these findings to build very expensive decision aids for battle commanders in the field. Unfortunately, most of the aids were disappointing. No one would use them" (G. Klein 1999, 7). One reason for this is likely to be the fact that decision making in war carries a high price for error (Alberts, Garstka and Hayes, et al. 2001, 37). This, combined with the intrinsic uncertainty and changeable availability of information, makes war a hostile environment to conduct decision making within.

[redacted]

The Military Decision-Making Process (Department Of The Army 1997, 5-1)

Military decision making requires both closed and open viewpoints. Information superiority, when effectively translated into superior decisions, achieves decision superiority. To enable this decision making is moving down the traditional command and control chain. Decision aids have been implemented before; anecdotal evidence indicates they were poorly received. The developed decision support system must support how military decision makers actually make decisions to help realise decision superiority. The U.S. Army's decision making process, the Military Decision Making Process, is the first considered.

2.3.2 Military Decision Making Process

The Military Decision Making Process (MDMP) is the U.S. Army's decision making process. It is an analytical approach to problem solving:

[redacted]

The Military Decision-Making Process (Department Of The Army 1997, 5-1)

The MDMP is a tool that assists the commander and their staff members to examine the situation, develop alternative courses of action and reach logical decisions. It is a scalable approach, such that when there is sufficient planning time and staff support, a comprehensive evaluation of numerous friendly and enemy courses of action can be conducted. In more time constrained situations the products created previously (when time and staffing allowed thorough evaluations) can be utilised.

The MDMP utilises doctrine to ensure a consistent understanding of terms and symbols used in the process, which also helps to maintain consistent situation assessment. The complete MDMP, whilst a time consuming process, analyses multiple friendly and enemy courses of action to ascertain the optimal friendly course of action (COA). Utilising this process helps ensure the integration, coordination and synchronisation of the operation, whilst preventing oversights. The use of the complete MDMP results in a detailed set of plans/orders based in doctrine to ensure consistency of meaning. The complete MDMP process is shown in Figure 8.

Figure 8 – The Military Decision Making Process (Department Of The Army 1997)

The MDMP has seven steps, with each step contributing to the next. A brief synopsis of each step presented in Appendix D: The Military Decision Making Process.

Figure 9 – Staff Inputs and Outputs in the MDMP (Department Of The Army 1997)

In time constrained environments omitting steps of the MDMP is not a solution. Rather, four primary techniques are suggested to save time.

1. Increase the commander's involvement, allowing him to make decisions during the process without waiting for detailed briefings after each step.
2. The commander becomes more directive in his guidance, limiting options. This saves the staff time by focusing members on those things the commander feels are most important.
3. The commander limits the number of COAs developed and war-gamed, in extreme cases he can direct that only one course of action be developed. The goal is an acceptable COA that meets mission requirements in the time available, even if it is not optimal.
4. The fourth and most time saving technique is maximizing parallel planning. Although parallel planning is the norm, maximizing its use in a time-constrained environment is critical. In a time-constrained environment, the importance of warning orders increases as available time decreases. A verbal warning order now is worth more than a written order one hour from now. The same warning orders used in the full MDMP should be issued when the process is abbreviated. In addition to warning orders, units must share all available information with subordinates as early as possible.

The MDMP is a *normative* and *prescriptive* decision making process. In this research we seek a *descriptive* model of military decision makers. The most widely recognised descriptive military descriptive model is the OODA Loop as described in the next subsection.

2.3.3 Observe, Orient, Decide and Act Loop

Col. John Boyd, a United States Air Force (USAF) military strategist, developed the OODA decision making loop. OODA refers to four overlapping and interacting processes: Observe, Orient, Decide and Act. This loop is reproduced in Figure 10.

[redacted]

Figure 10 – John Boyd's OODA Loop (Boyd, The Essence of Winning & Losing 1995)

Boyd was a very good fighter pilot and had the nickname "40-Second Boyd" because of his ability to defeat any opponent in aerial combat within this time limit. From his extensive combat experience he developed the OODA loop as a way of explaining to USAF trainees how to defeat the enemy in aerial combat.

[redacted]

John Boyd, Patterns Of Conflict (Boyd 1986, 5)

Boyd presents this OODA loop as occurring at tactical and strategic levels (Boyd 1986). War is conducted at three descending hierarchical levels: Strategic, Operational and Tactical. Decision making occurs at all three levels. (U.S. Department Of Defense 2006)

- [redacted]

[redacted]

(U.S. Department Of Defense 2006, 509)

- [redacted]

(U.S. Department Of Defense 2006, 391)

- [redacted]

(U.S. Department Of Defense 2006, 526)

The OODA Loop is a descriptive model of military decision making, originally developed by John Boyd to describe aerial combat but applicable across the military domain at the tactical level. Some recent concepts about how war is conducted is starting to speed up this process due to the increased proliferation of information, as discussed in the next subsection.

2.3.4 Network Centric Warfare

Network Centric Warfare (NCW), or Networked Enabled Capability (NEC) as it is known in the UK, provides the theory for warfare in the current Information Age. NCW is essentially about evolving military organisational behaviour to generate a tactical advantage from the effective networking of a war fighting enterprise. This evolution can be typically characterised as the ability of geographically dispersed entities to create a high level of shared battlespace awareness. This provides an inherent information superiority position which can be exploited through self-synchronisation and other tenets of NCW theory (Alberts and Hayes 2003, 98). The shift towards these tenets through a development of situational awareness and command and control is shown in Figure 11.

Figure 11 – NCW Maturity Model (Alberts and Hayes 2003, 109)

The rate of conversion from a position of information superiority to action is known as speed of command. Traditionally speed of command can take hours because of the time it takes to process, correlate, interpret and then act on the information through the chain of command. NCW principles combined with command and control (C2) processes could reduce the speed of command significantly. Hence, NCW provides the concepts to obtain the maximum tactical advantage from position of information superiority. NCW provides a broad organisational influence as it is transparent to mission, force size and geography. In addition NCW has the potential to contribute to the coalescence of the tactical, operational and strategic levels of war, which were outlined in section 2.3.3. This coalescence supports the current drive in military organisations to move towards the more flexible system of systems that modern conflict demands. In summary, NCW is not limited to technology and could be considered to be an emerging military response to the Information Age. The U.S. Marine Corps Distribute Operations concept provides a good example of the implications of the tenets of NCW on military decision making. Distributed Operations is a U.S. Marine Corps operational concept of how to meet emerging asymmetric warfare threats. The premise of Distributed Operations is to utilise the tenets of concepts such as Networked Centric Warfare to allow the decentralisation of informed decision-making down the traditional command chain to squad level, as shown in Figure 12.

[redacted]

Figure 12 – The Proliferation of Decision-Makers (Hagee 2005)

Rather than prescribing courses of actions the subordinate commander is empowered to develop their own, guided largely by commander's intent (Hagee 2005). Commanders intent is a concept that has its roots in the French Revolution and General Napoleon Bonaparte's method of conducting war. After he defeated the Prussians at Jena-Auerstadt in 1806 the Prussian army developed the *Aftragstaktik* philosophy, best translated as mission-orientated command (Widder 2002). The foundation of this philosophy was the concept of commander's intent, which is a description of the commanders desired outcome/end state that subordinates would work towards without being prescribed how to achieve it (Shattuck 2000).

[redacted]

Widder, Auftragstaktik and Innere Führung (Widder 2002, 5)

Today *Auftragstaktik* is inseparably linked with *Innere Führung*, which is the commitment of a soldier to moral and ethical standards. Essentially *Innere Führung* is the German military's corporate culture (Widder 2002).

The use of commanders intent as a cultural philosophy combined with concepts of Distributed Operations aims to break down the hierarchical levels of war into a flatter, more agile methodology for conducting war. At the centre of all of these concepts is the proliferation of decision makers at the lower ranks to enable faster response times and more adaptable approaches to the changing environment of the battlespace. This proliferation of decision makers is supported through the shared awareness that enables self-synchronisation as shown in Figure 11. This shift in military operations, both in execution and in tempo, marks a changing point in military decision making. The next subsection explores the literature to understand what these changes may entail for future military decision makers.

2.4 Future Trends in Military Decision Making

There are alternatives to the classical decision making theories that were presented in subsection 2.2. These alternatives mostly emanate from descriptive studies of decision making in naturalistic environments. Naturalistic Decision Making is the description of proficient decision making strategies that emphasise recognitional as opposed to analytical processes (Klein and Calderwood 1991, 1018). A leading theory of Naturalistic Decision Making is Klein's Recognition Primed Decision (RPD) strategy, with its inception in (Calderwood, Crandall and Klein 1987), extension in (Klein, Calderwood and MacGregor 1989, Klein and Crandall 1996) and collated into a full description in (G. Klein 1999). Klein's model of decision making was based on investigations of "the decision making strategies used by experienced personnel in operational setting, such as urban fire fighters, wild land fire fighters, tank platoon commanders, paramedics and design engineers" (Klein, Calderwood and MacGregor 1989, 463). Klein's research found that unlike traditional comparative theories the people he observed and interviewed did not make choices between alternatives. Rather, the decision makers acted and reacted based on "prior experience, planning, monitoring and modifying plans to meet specific constraints" (Klein and Calderwood 1991, 1020). There was no consideration of an optimal choice, which was seen by the decision makers as potentially paralysing – the so called 'paralysis by analysis', rather workable actions were chosen. At the core of this process was the decision makers ability "to recognise and appropriately classify a situation to generate a typical way of reacting" (Klein and Calderwood 1991, 1020). These decision strategies have been described in a recognition primed decision (RPD) model as shown in Figure 13.

Unlike the comparative methods discussed in subsection 2.2 the model proposed by Klein has more in common with Simon's satisficing (1979) which seeks the first option that works as opposed to optimising where the best option is sought that is harder to achieve (G. Klein 1999, 20).

Figure 13 – Recognition Primed Decision Model (Klein, Calderwood and MacGregor 1989, 464)

The implications for the military based on this model are considerable when compared to the MDMP as presented in subsection 2.3.2. The MDMP and other traditional step-by-step models have been described by Schmitt and Klein (1999, 510) as "inconsistent with the actual strategies of skilled planners, and they slow down the decision cycle. As a result, the formal models are usually ignored in practice, in order to generate faster tempo." From Boyd's OODA loop as shown in subsection 2.3.3 any slowdown of the decision cycle is a potential weakness, hence in line with NCW a faster tempo is sought. There is considerable evidence in the literature to support Schmitt and Klein's (1999) claim that the formal models are usually ignored in practice (McLamb 2002, Bushey and Forsyth 2006, Thunholm 2006).

RPD has been proposed by several authors as an enabler towards a better decision making process within the military (Ross, et al. 2004, Schmitt and Klein 1999, Bushey and Forsyth 2006, Thunholm 2004). In line with the stated objectives of this research to enable the developed Decision Support System (DSS) to be adopted most easily by the military by conforming to existing decision making models, the RPD will be adopted as the model of current and future military decision making for the purposes of this research.

This research has found the Military Decision Making context relates most closely to RPD Making which is a leading theory of Naturalistic Decision Making; hence RPD was selected as the model of choice for this research. It was felt to be beyond the remit of this PhD to conduct an in depth survey of decision making approaches and techniques due to the scope and depth of this subject area. The literature survey was thus limited to differentiating between classical decision making theories to situate the choice of RPD in the military context of this research.

2.5 Combat Search And Rescue Missions

To consider how to support effective military decision making a practical application was required. To ensure that the research would be valuable and applicable to the SEAS DTC a military mission was an obvious context to use. The majority of combat missions were deemed unsuitable by the researcher due to security issues, especially with access to information. The Combat Search And Rescue (CSAR) mission was chosen as the initial practical application as there is a quantity of literature (including United States military doctrines) available within the public domain. An initial review of this literature indicated that the mission involved decision making in a variety of situations at various levels of authority, making it suitable for this research. CSAR is also an approved SEAS DTC vignette for research. Whilst the overall outcome of the research will be focused on CSAR, the less specific outcomes should be valid and applicable to a number of other mission types.

A United Kingdom focus was sought for the CSAR mission but was not possible due to the very limited amount of information available from the UK military. A literature search found that the United States made CSAR related documentation freely available. For this reason the CSAR aspect of this research has a distinct United States basis simply because of the unrestricted availability of information, particularly military doctrines. A definition for CSAR was initially sought from these sources and a number of definitions were found within the available literature; a small selection of which is presented in Table 1.

Table 1 – CSAR Definitions

Literature	**Definition**
Department of Defense Dictionary of Military and Associated Terms (U.S. Department Of Defense 2006)	A specific task performed by rescue forces to effect the recovery of distressed personnel during war or military operations other than war.
Joint Doctrine Encyclopaedia (U.S. Department Of Defense 1997)	
Combat Search and Rescue	CSAR is a specific task performed by rescue forces

(United States Air Force 1998, 1)	to effect the recovery of distressed personnel during major theatre war or military operations other than war (MOOTW).
Joint Tactics, Techniques, and Procedures for Combat Search and Rescue (U.S. Department Of Defense 1998)	Combat search and rescue (CSAR) encompasses reporting, locating, identifying, recovering, and returning isolated personnel to the control of friendly forces in the face of actual or potential resistance.

Note that all of these definitions have a human focus. Due to the SEAS DTC project that this research supports focusing on (semi) autonomous systems (S/AS), it seemed clear that the generic CSAR functional model would need to orchestrate CSAR missions consisting of both human and platform (non-human) targets. Therefore, the following definition was adapted from existing definitions for the context of this CSAR modelling work:

"CSAR is a specific task performed by rescue forces to effect the recovery of assets isolated in hostile territory."

There are a set of assumptions that accompany the above definition, which are:

- "Hostile territory" refers to an area where opposing forces have the intent and capability to effectively oppose or react to recovery operations and/or threaten the isolated asset.
- The "territory" could include land, sea and littoral rescues, but space (as presently considered a demilitarised zone) and air (deemed unlikely) are excluded.
- An "asset" includes humans, platforms and data.
- "Isolated" refers to an asset when it becomes separated from its operational unit and is in danger of being captured.

To provide further context a brief history of CSAR is now presented.

2.5.1 A Brief History of Combat Search And Rescue

[redacted]

That Others May Live (Taylor Jr. 1967)

The first known aerial rescue was conducted by the French during the Franco-Prussian War in 1870 when observation balloons airlifted some 160 wounded soldiers away from the Prussian artillery (Taylor Jr. 1967, 61). CSAR missions started during the Second World War when the air forces for both Britain and Germany started to mount dedicated missions to rescue downed air crews from the heavily trafficked English Channel. Britain set up a number of Air Sea Rescue (ASR) squadrons which were dedicated to this task. At this time of course no helicopters were in service and so all rescues were conducted using aircraft such as the Supermarine Walrus, the Avro Anson and the Westland Lysander (Evans 1999, 9). Innovations such as the use of radar helped identify survival dinghies at longer ranges and in poorer weather conditions than previously possible. The range of these rescue missions was increased by using larger planes, such as the Consolidated Catalina and the Short Sunderland flying boat. By the end of the Second World War the British and American rescue forces had recovered 5721 airmen from the seas around the UK and a further 3200 airmen worldwide (Morgan 2003, 21). The most important technological change for CSAR missions was the advent of the helicopter at the end of Second World War. Without the limitations of having to land to rescue personnel thanks to the ability of the helicopter to hover and the use of winches to load and unload, the helicopter became the favoured CSAR recovery platform. In March 1946 the United States of America (USA) established its own Air Rescue Service (ARS) within Air Transport Command to provide rescue cover for the continental USA (Evans 1999, 14, National Musueum of the USAF n.d.). The scope of the ARS had expanded by 1949 to cover all of the world's transport routes and also included war zones which involved US or UN personnel. In 1966 the ARS was re-designated the Aerospace Rescue and Recovery Service (ARRS). This peace time organisation was transformed through necessity by the Vietnam War into a specialised combat rescue organisation. The ARRS crews saved 4,120 lives during Vietnam, 2,780 in combat situations (Evans 1999, 19). Whilst the equipment and techniques have been much improved since the Vietnam conflict the basic mode of operation has remained the same, with an emphasis on five key interrelated stages (Joint Chiefs of Staff 1998, ix): awareness and notification, situation assessment, mission planning, execution and mission conclusion.

2.5.2 An Example Combat Search And Rescue Mission

[redacted]

Aerospace Rescue and Recovery Service Code (Evans 1999, 19)

For the benefit of the reader a brief synopsis of a CSAR mission is provided below: the 1991 rescue of Lt Devon Jones, an F-14 pilot downed near Baghdad, Iraq, during the Gulf War. This example is given as it will be used later in this thesis for a historical case study. This synopsis, culled from (Evans 1999, 133-136, National Museum of the USAF n.d., Pokrant 1999, 31-32) is intended to give the reader some insight into how CSAR missions are conducted. This was the first rescue of a downed airman during Operation Desert Storm.

Lt Devon Jones and his Radar Intercept Officer, Lt Larry Slade, were tasked with flying their F-14A+ Tomcat, call-sign “Slate 46”, on an armed escort mission to protect an EA-6B Prowler aircraft around the airfield at Al Asad, Iraq. At 6.05am whilst returning to their base, approximately 30 miles from Baghdad, Slate 46 was hit by a SA-2 SAM (Surface to Air Missile) and both Jones and Slade ejected at approximately 10,000 feet. Slade used his AN/PRC-112 survival radio whilst parachuting down to call in a “mayday” to the orbiting AWACS airplane. The AN/PRC-112 survival radio provides Army Search and Rescue (SAR) personnel with the capability to perform combat search and rescue (CSAR) missions of downed aircrew personnel. As Jones and Slade descended by parachute they were separated in the clouds and darkness and landed separately, some distance apart.

Jones dug in and camouflaged himself. Meanwhile a MH-53J ‘Pave Low’ helicopter scrambled to crash site. Overhead Iraqi MiG-23 fighters appeared and were engaged by two USAF F-15Cs on Rescue Combat Air Patrol (RESCAP), the MiG-23’s retreat. At approximately 10:30 Jones hides from what he thought were enemy fighters, but which were actually the two F-15Cs on RESCAP. At approximately 10:30 Slade was discovered and detained by Iraqi forces. MH-53J ‘Pave Low’ helicopter returned to Arar to refuel, returning immediately to continue the search. The ‘Sandy’ A-10A made contact with Jones via radio on SAR frequency. Jones guided the ‘Sandy’ A-10A to him and the ‘Sandy’ A-10A located him exactly using its Inertial Navigation System (INS). Iraqi MiG-23s moved into the area and were engaged by two USAF F-15Cs on RESCAP, the MiG-23’s retreated. ‘Sandy’ A-10’s sanitised the area, MH-53J ‘Pave Low’, returned from refuelling, waited on the ground. MH-53J ‘Pave Low’ moved in to rescue Jones, and spotted an Iraqi truck en route to Jones’ position which was engaged by a ‘Sandy’ A-10A. MH-53J ‘Pave Low’ landed near the destroyed Iraqi truck, Jones broke cover and boarded the helicopter as shown in Figure 14, which returned to base. ‘Sandy’ A-10A’s stayed on station to assure the MH-53J’s safety.

Figure 14 – The Moment Lt Devon Jones is Rescued. (United States Air Force 1998, 2)

With a historical example of a CSAR mission given we now consider the general characteristics of such missions, which is the focus of the next subsection.

2.5.3 General Combat Search And Rescue Mission Characteristics

The example given in the previous subsection 2.5.2 helps highlight the general characteristics of a CSAR mission. These characteristics are:

- the relatively small number of dedicated CSAR systems used (four) and the comparatively larger system of systems within which they operated (within the military force deployed during the conflict).
- the overall time constraints of the mission (the MH-53J scrambled immediately after notification to the crash site)
- the short response time (four and a half hours) for an appropriate SoS to be formed (the "rescue package")
- the high level of collaboration and interoperability required between the various sub-systems to effectively respond to the changing, unpredictable environment within which they were operating.

2.6 Chapter Summary

This Chapter has considered decision making under uncertainty and seeks a descriptive model of military decision makers for which a decision support system can be developed. Two main decision making theories from economics are the *Expected Utility Theory* which states that people maximise utility as opposed to reward and *Prospect Theory* which characterises the general behaviour of decision makers with the terms Loss Aversion and Diminishing Sensitivity. Both of these theories indicate that humans are innately irrational decision makers liable to be subject to a number of decision making fallacies. Note that this does not specify how the decision makers actually make the decision, rather it characterises the likely features of the decision made. A DSS would be useful to help negate these decision biases.

Aware of these human limitations of decision making the next consideration was of how decisions are currently made within the military. The role of information was considered and its increasingly important role in redefining how superiority over an enemy can be defined in terms of decision making. This found that military decision making is typically time constrained and highly stressful which limit human decision making and can negate any informational advantage. A DSS which could alleviate these effects and maintain any informational advantage would be useful.

The history of decision making in military operations was traced and the Military Decision Making Process and Observe, Orient, Decide and Act Loop were discussed. Emerging tenets such as Network Centric Warfare (NCW) and its impact on situational awareness and command and control were presented. NCW is not limited to technology and could be considered to be an emerging military response to the Information Age. The proliferation of decision makers leads to alternatives to the classical decision making theories. These alternatives mostly emanate from descriptive studies of decision making in naturalistic environments. Naturalistic Decision Making is the description of proficient decision making strategies that emphasise recognitional as opposed to analytical processes (Klein and Calderwood 1991, 1018). A leading theory of Naturalistic Decision Making is Klein's Recognition Primed Decision (RPD) strategy which has been selected as the most representative model of how military decision makers actually make decisions. This will be the decision making model that this research will seek to support.

The particular military mission that the DSS will address is the Combat Search And Rescue (CSAR) mission. This mission was chosen as it is a time sensitive reactive mission requiring a virtual SoS to be rapidly assembled to conduct the mission. With a practical application for this research established the next chapter considers the development of the DSS.

Chapter 3

An Architecture of an Approach for a Decision Support System

Outline of Chapter

This chapter presents the initial development of a Decision Support System (DSS) to meet the requirements established in the first two chapters of this thesis. This chapter seeks to prove that a common set of metrics exist that can be used to compare missions to systems of systems, which will provide the foundation for the DSS developed in the following Chapter 4.

Research Contributions of Chapter

From the previous chapter(s):

*This **Research** has established a **Problem** and **Objectives** to develop an **Architecture** of an **Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

This chapter introduces:

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics**. This set of common characteristics which includes **Functional** and **Non-Functional** characteristics, was used to inform the decision maker. The non-functional characteristics were characterised by **Attributes** which are composed of **Secondary Attributes** which, in turn, are composed of **Factors**. The functional characteristics were captured in a **Generic Functional Model** for combat search and rescue missions.*

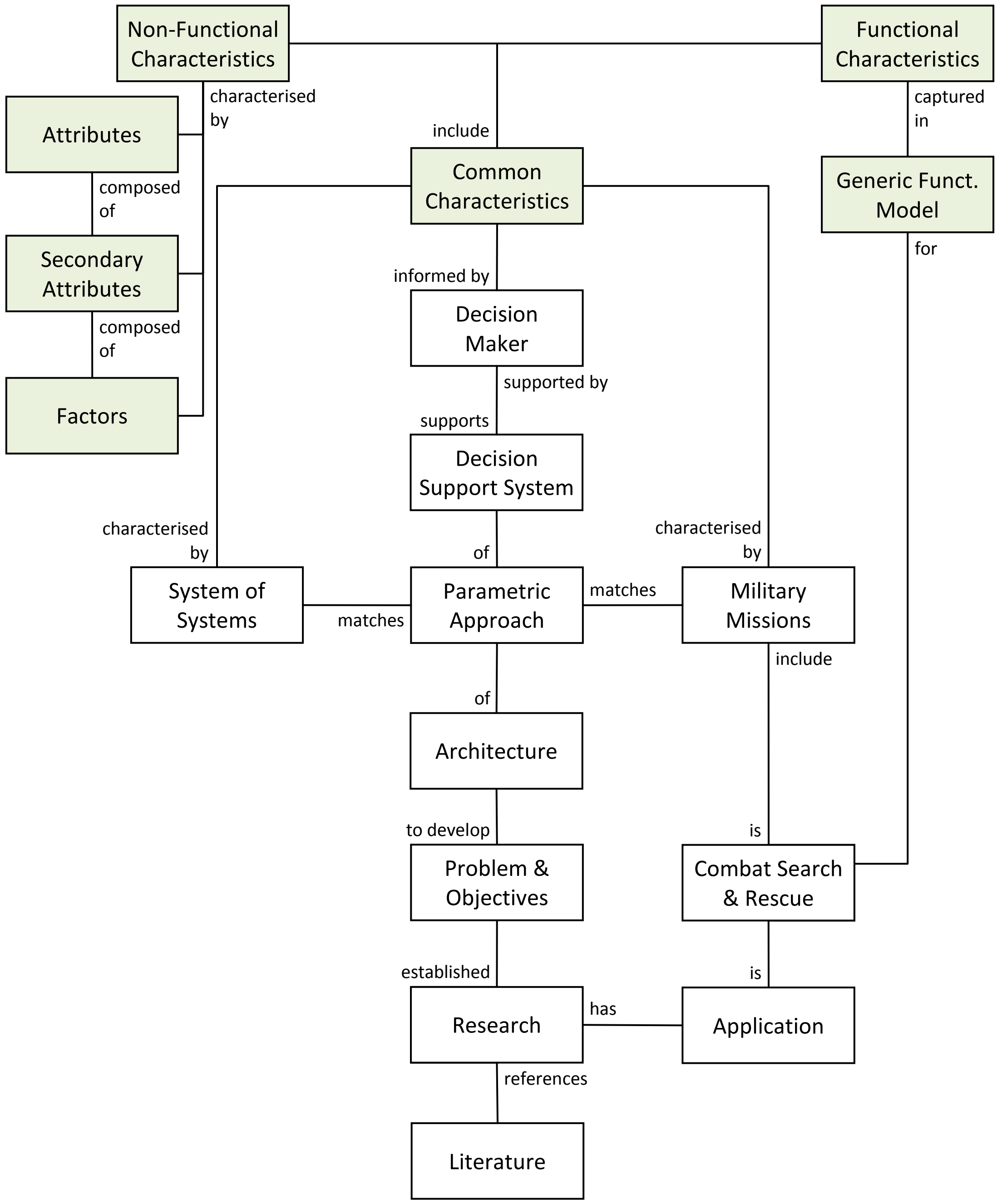


Figure 15 – Logical Model of Research Contributions of Chapter 3

3.1 Characterising Missions and Systems

The decision of which SoS to use for a particular CSAR mission is critical, as a poor selection or structuring of the SoS could result in the deployed SoS being incapable of achieving the mission objectives. However, CSAR missions in general cover a wide range of physical and tactical scenarios all of which have different objectives and hence it is necessary that any methodology used should allow differing levels of capability to be defined for each specific CSAR mission. This leads to an awareness that differing levels of capability can be defined for specific missions, which requires an SoS to be more than just functionally and structurally fit for a particular mission. It also requires the SoS to have appropriate operational characteristics, such as adaptability and interoperability, i.e. some of the so-called 'ilities'. To develop the architecture for an approach for the DSS required to address the research problem previously outlined in section 1.3 a common set of characteristics is required which can be used to profile both the mission and the available SoS to allow them to be matched.

The approach for the decision support system is to have sets of common characteristics from which comparative profiles for the mission and available SoS can be created that inform and support a decision maker, as illustrated in Figure 16. These profiles are implemented by the inputs on either side of the diagram in Figure 16. Note that at this stage we are seeking to list out the components of the approach and establish relations to the solution. The approach will be fully developed in the next Chapter where the architecture will be implemented.

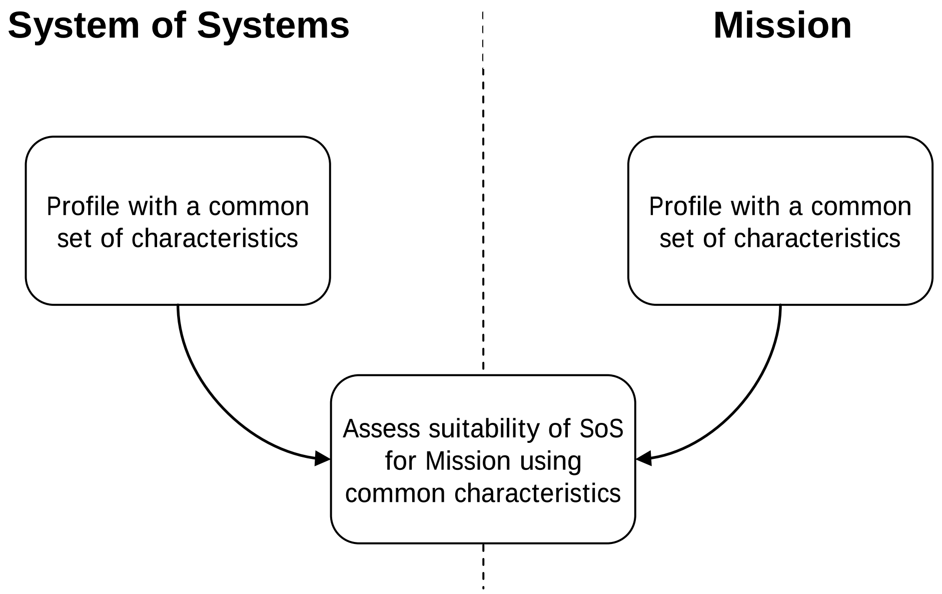


Figure 16 – An Initial Architecture of an Approach for a DSS

A profile is defined by the Oxford English Dictionary as:

[redacted]

(Oxford English Dictionary 1989)

In the context of this research a profile is specifically defined as *a representation of the structured set of common characteristics of a mission or a system of systems*. The assessment referred to in the lower box in Figure 16 is, at this initial stage of development, envisioned to be a comparative assessment allowing a proposed SoS Profile to be compared to the mission's profiles and mismatches highlighted between the two. The assessment process will be refined and developed over the course of this chapter as the common characteristics are defined and in the next chapter when the architecture of the approach is realised in a prototype software tool.

These common characteristics can be used to profile missions and SoS from both open and closed viewpoints. The implementation of the architecture of the approach for a DSS will determine the actual viewpoints taken which will likely be based on the practical issues of realising the developed DSS approach.

The work in this chapter was conducted through group work with Professor Mike Woodhead, Dr Carys Siemieniuch, research associate Mr Nick Meese and another PhD student Mr John Cleveley. Professor Woodhead has extensive experience in both aerospace and defence while Dr Siemieniuch has extensive experience in organisational systems engineering. Subject matter experts from the military had been sought through the SEAS DTC but were unavailable for this exercise. The group work established the high level structure of the various characteristics which were then detailed by this PhD candidate before being reviewed and refined by the group. Three sets of common characteristics to profile missions and systems were identified:

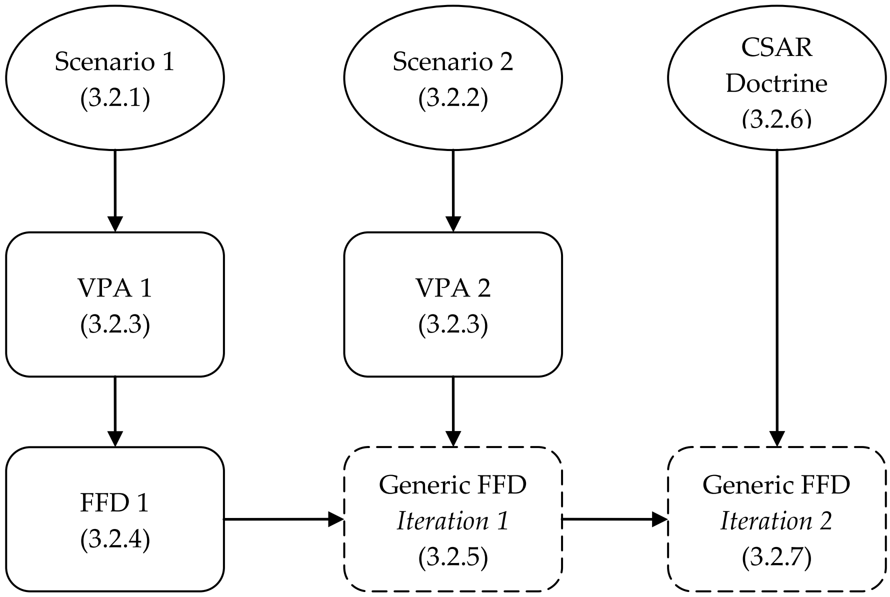
1. Functional (what the system needs to do).
2. Non-Functional (the operational characteristics of the system).
3. Physical Environment (the nature of the physical environment in which the system must operate).

Of these characteristics the functional and non-functional were considered by the group to be the most important and hence the most critical to ascertain whether a common set of both existed to found the decision support system on. These characteristics and the approaches to each to develop a common set of characteristics for each are explained in further detail over the next three subsections.

3.2 Functional Characteristics

To understand CSAR various analysis methods were used to build a visual representation of the mission, incorporating functional and non-functional characteristics. The foundation of this process was the consideration of two very different CSAR scenarios, described in the following sections. The scenarios were intentionally high level and non-specific to provide a wide scope for the top level requirements the analysis of them would produce. These scenarios were analysed in turn using Viewpoint Analysis (VPA) to identify stakeholders, identify key mission requirements (non-functional requirements) and to create a functional, solution independent view of the scenario. VPA is described in more detail in section 3.2.3. The first scenario's functions, generated by the VPA, were then associated to each other to form a functional structure using Functional Flow Diagrams (FFD). FFD are described in more detail in 3.2.4. This approach enabled an understanding, through visualisation, of how the CSAR system fitted together in terms of functions and their relationships. Note that while this exercise was conducted with knowledgeable civilians the aim was to produce a reasonable model which will later be verified against doctrine.

The FFD's were decomposed down several levels of functionality until it became apparent that the desired level of detail had been achieved. This created a functional hierarchy for a CSAR mission with its associated non-functional constraints. This FFD was then re-examined with the second scenario's VPA to produce a second FFD encompassing both scenarios. From this a generic CSAR FFD, or Generic Functional Model (GFM), was developed which is applicable to a variety of CSAR missions. This was achieved by stripping out the mission specific language in each scenario to leave a set of generic stages which could cover multiple CSAR mission types. The GFM is described in section 3.2.5. The GFM was verified functionally using available US CSAR doctrines, described in section 3.2.6. This entire approach along with the sections in which each stage is described in more detail is depicted in Figure 17.



(Numbers in parentheses refer to the subsection where they are discussed)

Figure 17 – Process Followed to Create a Generic Functional Model

The two scenarios used in this process are described next.

3.2.1 Scenario 1: Downed Airmen Behind Enemy Lines

In this scenario a fighter aircraft has been hit by a surface to air missile over enemy territory in the desert. The pilot managed to eject and has landed in a hostile area. It is imperative that the pilot is found before she becomes captured by enemy forces and is used for propaganda purposes. A signal from the ejector seat beacon had been detected, limiting the initial search area to about $10km^2$. The area where the ejector seat was detected was quite remote and there was unlikely to be any enemy forces in the immediate area until they could move troops in from nearby bases. This left a small time frame to rescue the pilot in before the enemy moved in. If the enemy did manage to quickly move a forward team into the area, it was likely to consist of lightly armoured vehicles and conscript soldiers. The downed aircraft had sensitive data and equipment on board that needed to be retrieved or destroyed. A diagram of this scenario is shown in Figure 18.

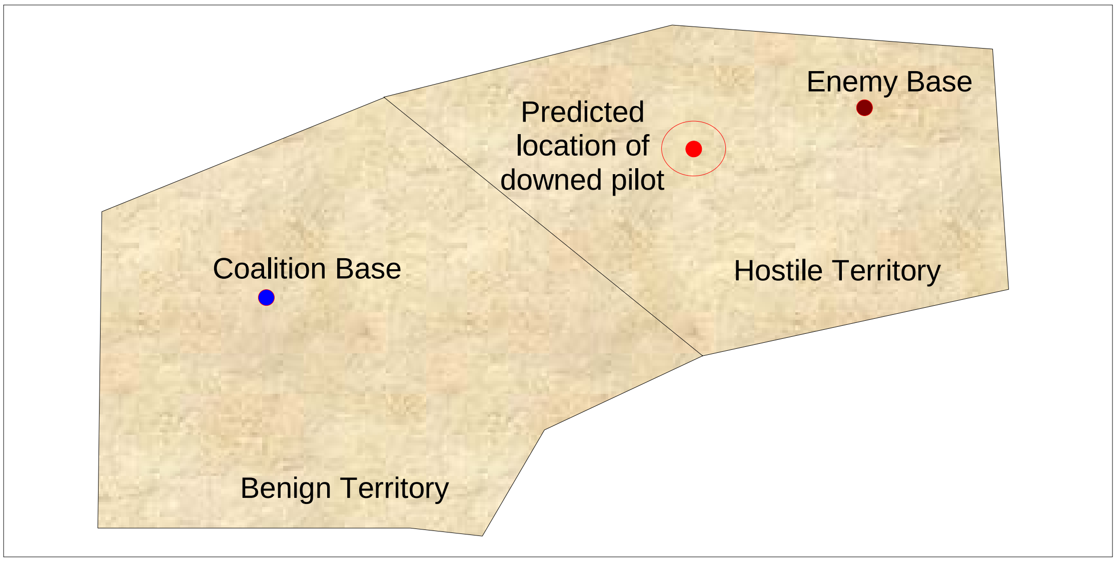


Figure 18 – Scenario 1 Map

3.2.2 Scenario 2: Lost Unmanned Autonomous Vehicle

In this scenario communication had been lost with an unmanned, autonomous water borne vehicle in hostile coastal waters. It was presumed that the vehicle had broken down. The vehicle's last recorded position was known, but due to local currents the vehicle may well have drifted significantly from that position. If the vehicle was found by the enemy it could cause significant political problems, therefore it was important that the vehicle was either recovered or destroyed without being detected by hostile forces. The enemy had significant radar coverage and intelligence indicated that the enemy had coastal patrol boats in the vicinity. A diagram of this scenario is shown in Figure 19.

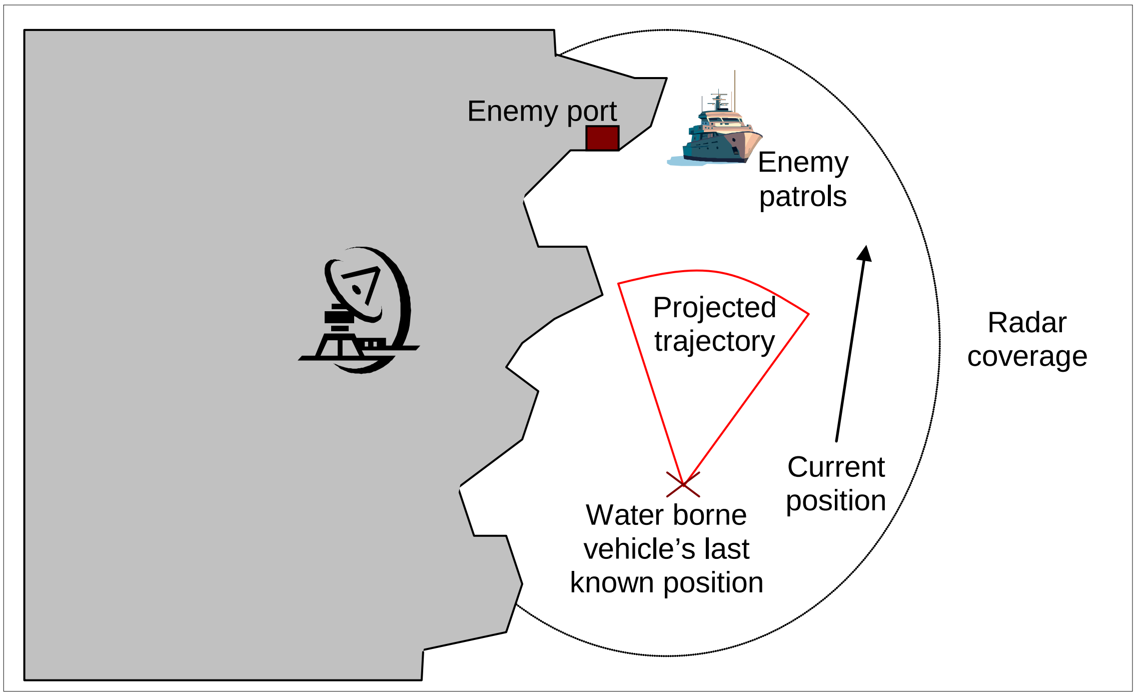


Figure 19 – Scenario 2 Map

3.2.3 View Point Analysis

View Point Analysis (VPA) is a semi-structured analysis process through which a set of functional and non-functional requirements can be generated (Burge and Woodhead 2006). VPA has its origins in software engineering and is a multi-perspective analysis which views problems from each of the stakeholders viewpoints. This can be done directly by consulting with the stakeholder or, in cases like this where they are unavailable by reconsidering the problem space from the identified stakeholders perspective. Identified stakeholders for the scenarios include the Rescue Target, Ministry of Defence, Civilians, Enemy Forces, the home nation Public (representing public opinion), Media, Suppliers, Friendly Forces, Coalition Members and Resistance Groups.

VPA was used to brainstorm the key elements of a CSAR missions in terms of their functional and non-functional requirements. This approach helped to gain a holistic view of CSAR by considering the perspectives of each of the identified stakeholders in turn. The two CSAR example scenarios, as discussed previously, were used as a basis for the VPA and, due to the diverse nature of the two scenarios, helped produce a wide set of top level requirements. The two scenarios were intentionally left quite open, lacking the usual detail found in a real military operation, to help increase the scope of the top level requirements. The functional and non-functional requirements produced by the VPA were solution independent, i.e. they did not dictate how the scenario will be implemented in terms of equipment, but rather in terms of high-level functions such as 'engage with enemy' or 'detect distress signal'.

The two scenarios were considered in turn. The first scenario's VPA diagram is shown in Figure 20. Note that the Operations bubble has been removed due to space constraints on this page, the contents are shown in the VPA diagram for Operations in Appendix E.2. For completeness this diagram is also repeated in Appendix E.1 to provide a complete set in Appendix E for the reader to follow through.

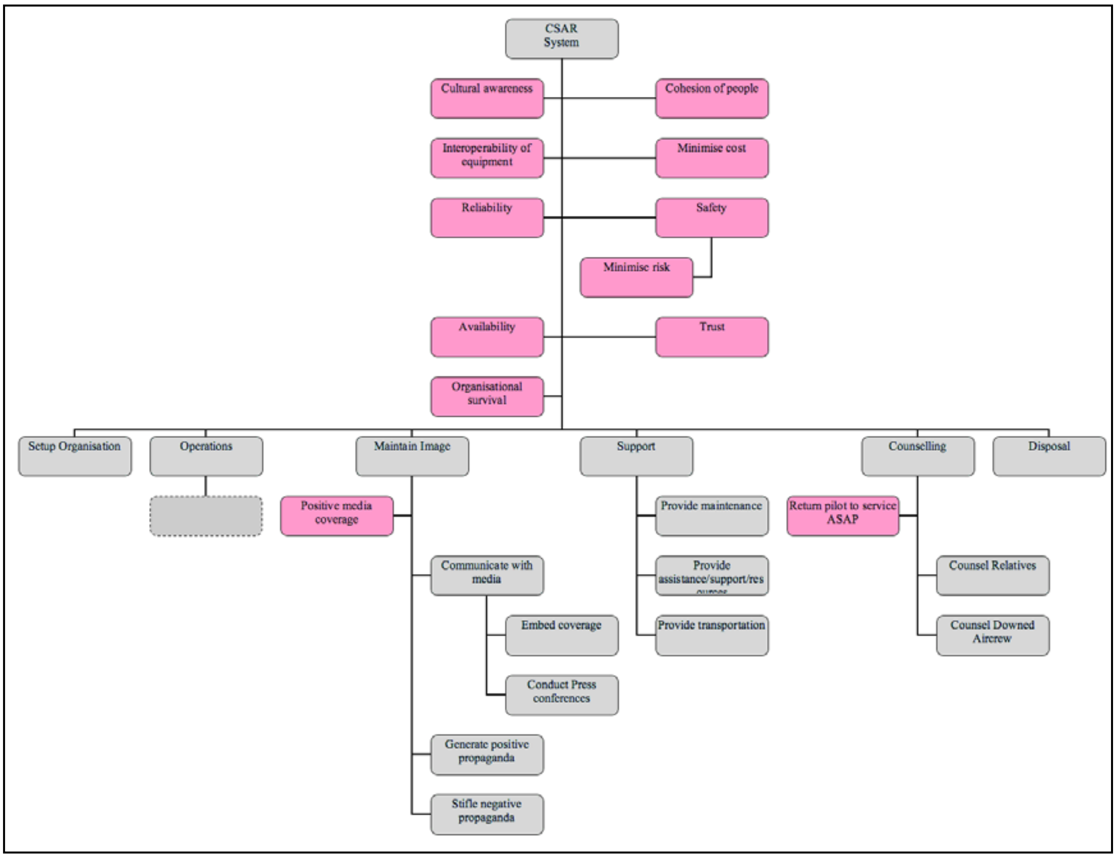


Figure 20 – Scenario 1 CSAR Mission Overview VPA Diagram

The second scenario's VPA diagram is shown in Figure 21. For completeness this diagram is also repeated in a larger form in Appendix E.2 to provide a complete set in Appendix E for the reader to follow through.

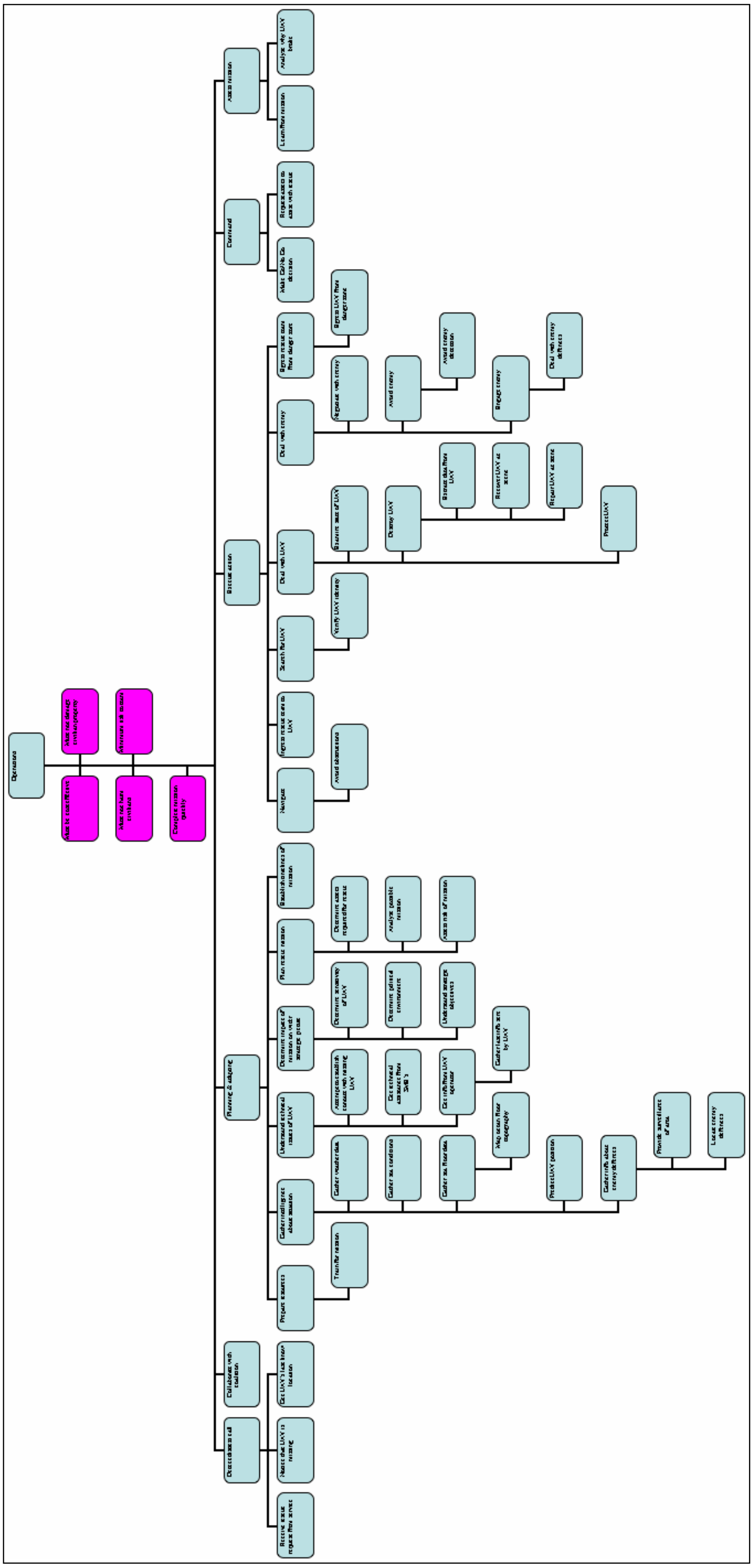


Figure 21 – Scenario 2 VPA

The two scenarios produced quite different VPA diagrams. Scenario one's analysis highlighted many soft issues related to the downed airman, including supporting them in the field and counselling upon their return. Consideration was also given to the propaganda issues resulting from mission success (the rescue of the airman) and failure (the loss or capture of the airman by enemy forces). Hence a consideration was given to *Maintain Image* which encompassed these facets of propaganda. The second scenario's VPA analysis highlighted more mechanistic considerations as the rescue was of an Unmanned Autonomous Vehicle (UAV), not a person. Hence the possible outcomes considered included recovering, repairing or even destroying the UAV. A number of key findings were made from the VPA of both scenarios, including:

- A consideration of the wider strategic picture is required for CSAR. A CSAR mission cannot be considered in isolation, there are always high level constraints such as international agreements.
- Common phases of CSAR were identified for both scenarios (which were later verified with US doctrine as described in section 3.2.6). Stripping out the mission specific language in each scenario left a set of generic stages which could cover multiple CSAR mission types.
- The scenarios highlighted common issues for both human and platform recovery. For example, the requirement to repair the unmanned asset in Scenario 2 triggered an additional requirement to administer first aid to a human asset in Scenario 1.

Six key functions were identified for undertaking a CSAR mission, these were: Setup Organisation, Operations, Maintain Image, Support, Counselling and Disposal. To provide a focus and some bounding to our work the Operations function was selected for further detailing, as this was the function that actually "performed" the CSAR mission. The VPA diagrams are shown in Appendix E, with the overview shown in Appendix E.1, the VPA for Operations in Appendix E.2 and the VPA diagram for the second scenario in Appendix E.3.

3.2.4 Functional Flow Diagrams

Functional Flow Diagrams (FFDs) are the product of a "top-down" structured functional modelling process and represent a system in terms of functions with inputs and outputs (Burge and Woodhead 2006). Producing FFDs allowed an understanding, through visualisation, of how the CSAR system fitted together in terms of functions and their relationships. An FFD diagrammatically represents functions (from the VPA) as ellipses with interlinking information flows. Each diagram is set at a particular functional level of the system, as determined by the hierarchy developed in the VPA. A functional ellipse can be explored in greater detail by creating a new FFD at a functional level below it. The process of producing FFDs helps to ensure consistency in the flows between functions and sub-functions. The FFD at the highest functional level is called a context diagram, which shows the whole system and its interfaces with the environment. The FFD for scenario 1 is shown in Figure 22 which depicts the overall CSAR system. Note the stakeholders used for the VPA are shown in the environment of the system (e.g. Civilians, Enemy Forces etc.).

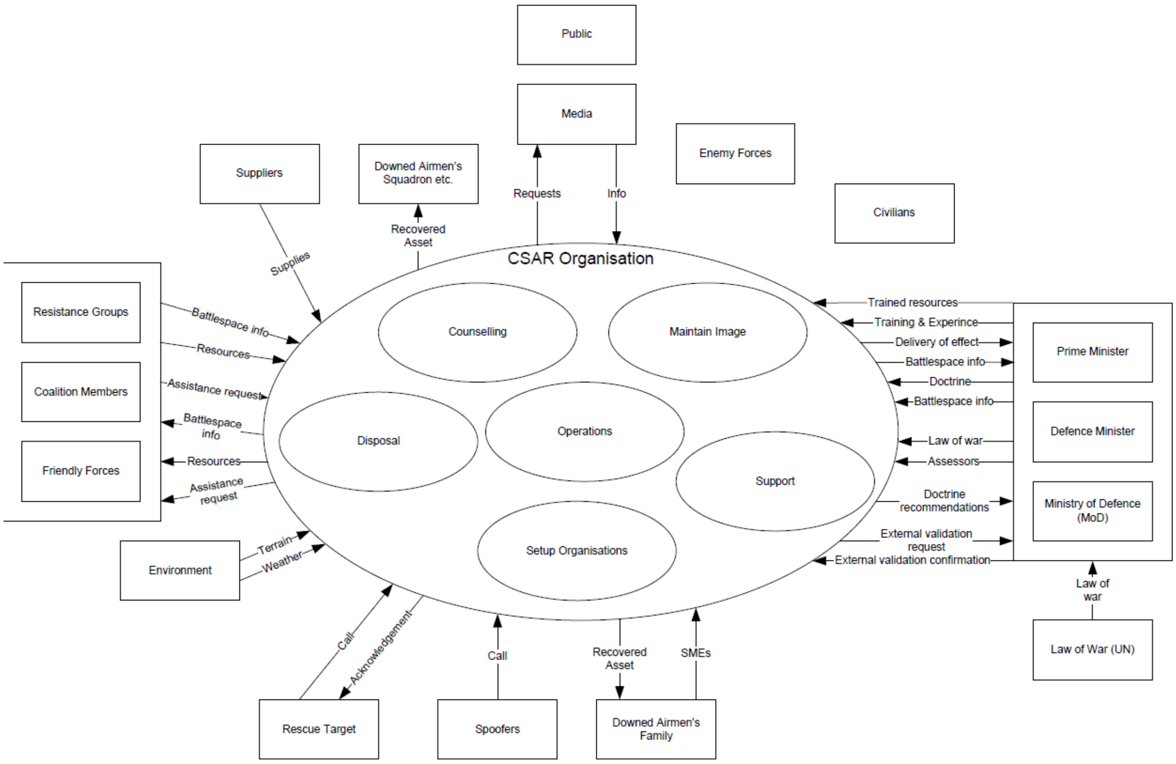


Figure 22 – Scenario 1 FFD Context Diagram

Subject matter experts from the military had been sought through the SEAS DTC but again were unavailable for this exercise. Initially the FFDs for the first scenario were produced, based on the VPA for that scenario. This model iteratively evolved over time as the team's thinking matured, aided by the rapid methodology of FFDs. When this first set of FFD's was produced the team defined the areas of interest within the CSAR system, allowing certain functionality to be drilled-down several levels until a desired level of detail had been achieved (e.g. for clarification of what was involved with a particular function). These FFDs were then re-examined with the second scenario's VPA to produce a second FFD encompassing both scenarios, the first iteration of a Generic Functional Model (GFM). An example of one of the FFDs produced is shown in Figure 23.

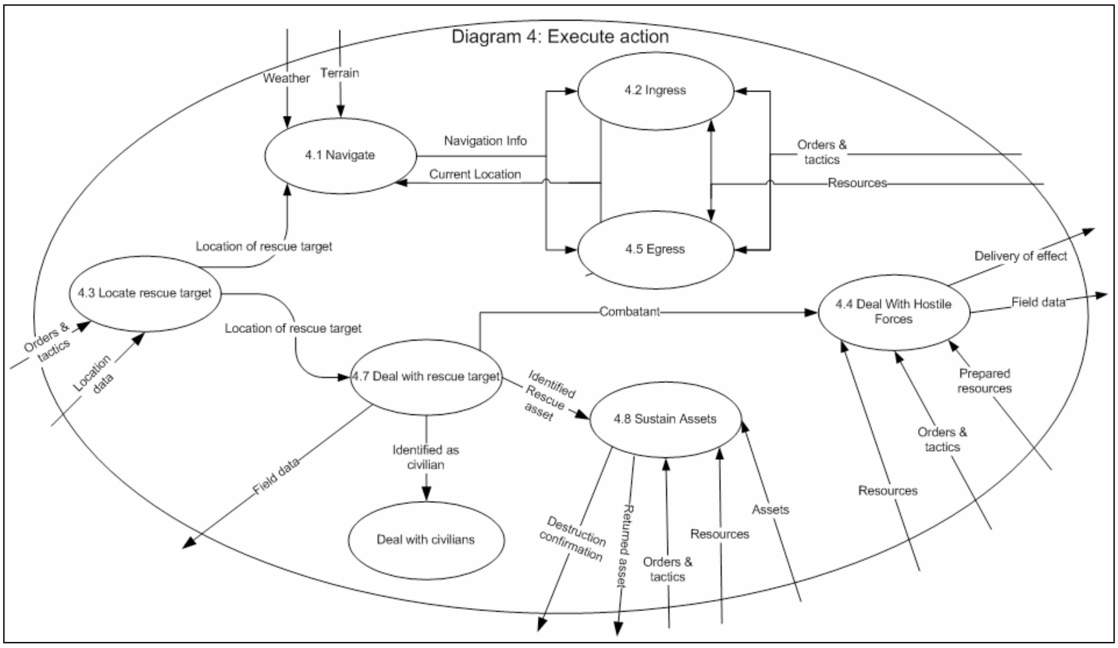


Figure 23 – The Execute Action FFD (from Generic FFD Iteration 1)

3.2.5 Developing a Generic Functional Model

A generic CSAR functional flow diagram or Generic Functional Model (GFM) applicable to a variety of CSAR missions was developed by considering the two scenario specific functional models. This was achieved by stripping out the mission specific language in each scenario to leave a set of generic stages which could cover multiple mission types. The GFM represents the current understanding of CSAR and can be continued to be developed by appropriate subject matter experts. Whilst the model is generic at the higher common levels it should be noted that as the model becomes more detailed it becomes more mission specific. The FFDs generated for the different CSAR scenarios showed that whilst every CSAR mission is unique at the more detailed levels, there were clear commonalities at the higher levels of abstraction. The GFM captures these high level commonalities, enabling it to support the majority of CSAR missions and hence providing a starting point for the development of the architecture(s) to be used for a particular CSAR mission.

Two iterations of the GFM are used for this research: the input of VPA 2 and then the verification against available doctrine as discussed in the next subsection. More iterations could and can be used, each iteration which considers another scenario will make the GFM more generic. However, as this is a wicked problem the GFM will never get to the “answer” only tend towards it. The GFM can therefore take additional information and incorporate it in an iteration, hence the GFM can be as good as the available knowledge allows. Note that there is no need to normalise the model after each iteration as it is normalised by inclusion in the GFM, no stand alone FFD for each iteration is required, only an evolution of the existing GFM. The full Generic Functional Model is shown in Appendix F - The CSAR Generic Functional Model.

3.2.6 Verification Of The Generic Functional Model

Once the GFM reached a reasonable level of maturity it was verified through functional analysis using available Department of Defence CSAR doctrines. This involved analysing the available doctrines to elicit the core functionality referred to in them, including the relationships between them where possible. This functionality was then compared to the GFM and, where the functional analysis of the doctrines differed from the GFM, the GFM was updated. A summary of the key Joint Operations doctrines are discussed below.

Joint Tactics, Techniques, and Procedures for Combat Search and Rescue (U.S. Department Of Defense 1998) describes the fundamentals of multinational operations, reviews multinational command relationships, discusses the considerations during the planning and execution of multinational operations and covers operational considerations.

Doctrine for Joint Combat Search and Rescue (U.S Department Of Defense 1996) describes CSAR responsibilities and command relationships, explains CSAR procedures and methods, outlines coordination and planning procedures, defines CSAR intelligence and support requirements and details CSAR capabilities of the services and special operations forces.

Joint Doctrine for Evasion and Recovery (U.S. Department Of Defense 1996) provides general evasion and recovery considerations, covers the moral, legal, and operational guidelines for evasion and discusses the philosophy and considerations of recovery.

This verification ensured that the GFM reflected the current military approach to the mission and incorporated what had been comprehensively tried, tested and learnt in the field. Differences between the GFM and literature were identified and improvements were made to the GFM to enhance its scope and functionality. The literature based verification was planned to be enhanced by consultation with subject matter experts but none were available within the research time.

For the purposes of the decision support system the aim of the GFM was to provide a realistic and useable model of CSAR operations. By following a process of model verification through literature based doctrine analysis and, ultimately consultation with subject matter experts, a realistic and useable model was developed for use in this research. This approach also helped ensure that there were no gaps in the developed model of required functionality. In practice it was envisaged that this model would be evolved as it is used by specialists and they add to or modify it as appropriate. The model may also evolve gradually over time as the functionality required by the mission shifts in response to the changing demands of warfare. Note that the iterative FFD approach combined with verification to the doctrine which introduced some normalisation of terminology means the functions identified in the VPA exercises would not necessarily map onto the FFDs and hence the generic functional model, which is the focus of the next section.

3.2.7 The Generic Functional Model

This initial stage of the research captured the functional requirements for conducting the majority of CSAR missions in a Generic Functional Model (GFM). This has been developed from a number of scenario-based case studies (including land and sea based missions) and verified using available doctrines. The GFM was developed by stripping out the mission specific language in each scenario to leave a set of generic stages which cover multiple CSAR mission types and which could be verified using available doctrines (U.S Department Of Defense 1996, U.S. Department Of Defense 1996, U.S. Department Of Defense 1998, United States Air Force 1998). US doctrines were considered because of their free availability in the public domain as opposed to the more restricted UK doctrines. The GFM represents a current understanding of CSAR missions which can be matured further by practitioners (Cleveley, Johnson, et al. 2007). While the model is generic at the higher common levels (e.g. “Execute Action”), the model becomes more detailed at the lower, more mission specific levels. This is illustrated in Figure 24 below.

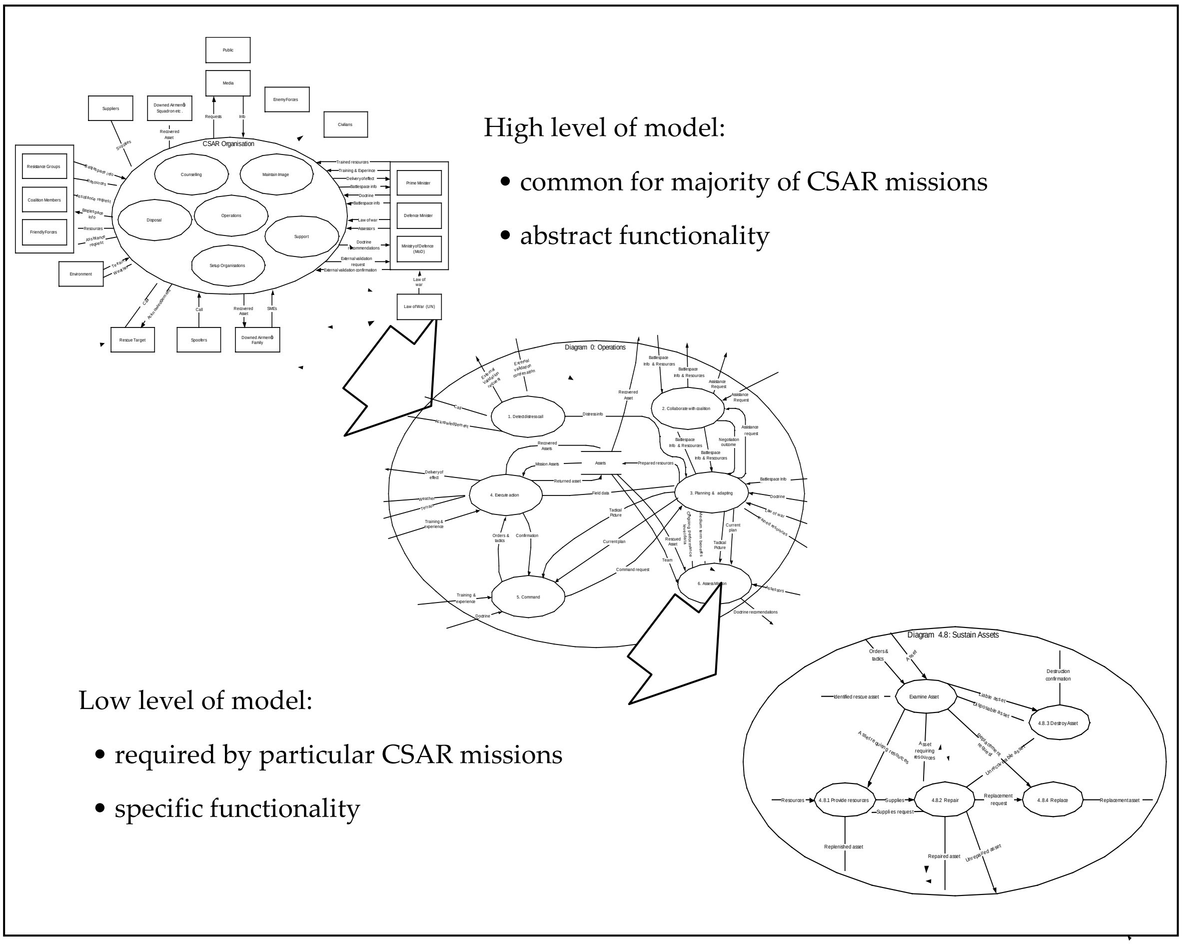


Figure 24 – The Hierarchical Structure of the GFM (Iteration 2)

The Functional Flow Diagrams generated for the different CSAR scenarios showed that whilst every CSAR mission is unique at the more detailed levels (as a wicked problem ought to be), there are clear commonalities at higher levels of abstraction. When a CSAR mission is initiated, the GFM can be used to identify the functions required to conduct the particular mission, noting that for a particular scenario some functions of the GFM may not be active. Understanding the required system functionality may contribute, along with military doctrine, to determining appropriate courses of action (COA) based around the purpose and objectives needed to complete a particular mission. A CSAR operation is a reactive mission, that is to say that the mission is only conducted when the event of someone/something going missing has occurred, which is inherently unpredictable. Given the nature of CSAR missions, they are generally time sensitive, requiring an immediate response to maintain a tactical advantage over the enemy and deny them time to organise and react. Whereas most military missions are planned some time in advance to a level of detail that allows a fair assessment of success, CSAR missions are not and once initiated by an event requiring a CSAR mission, are under severe time constraints. When such a mission is initiated it will be necessary to determine what resources (platforms, people, supplies etc.) are available to the CSAR organisation to conduct the mission. As stated previously, whilst the GFM provides a high level picture of what is required functionally to complete the mission, it will be necessary to assess and detail this with respect to the demands of a particular mission. The GFM is used to determine whether an architecture is suitable by matching it's functionality to that of the profiled GFM. Mismatches can be identified and will help inform the decision maker of the suitability of that architecture to conduct the mission.

3.2.8 Architecting Functional Characteristics into an Approach for a DSS

With a GFM developed which can be used to profile both missions and available SoS a consideration was given as to how it could be utilised in the overall concept of the architecture of an approach for a DSS as outlined previously in Figure 16. While the initial architecture used only *processes* it is now necessary to include *inputs*, *outputs* and *components*. *Inputs* and *outputs* are self explanatory and indicate the inputs and outputs of the architecture of an approach for a DSS. The *components* are data stores developed in this research which capture the generic characteristics of the missions and SoS and are used by the approach for a DSS to rapidly create specific characteristic profiles for particular missions and SoS. The first component developed was the GFM. The updated architecture of an approach for a DSS which utilises the developed GFM is shown in Figure 25.

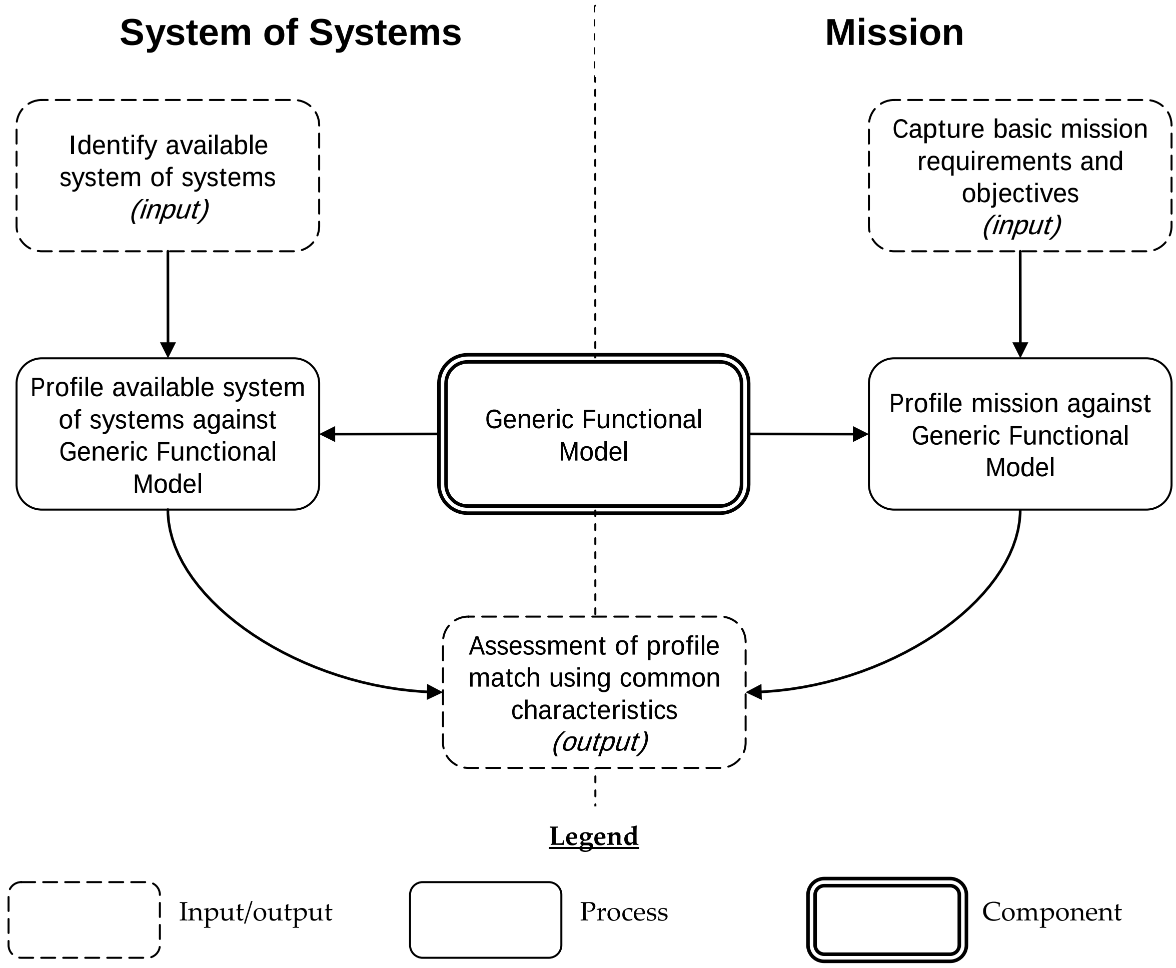


Figure 25 – An Architecture of an Approach for a DSS Utilising a GFM

With a common set of functional characteristics captured in the developed GFM the next stage was to ascertain if such a set existed for the non-functional characteristics.

3.3 Non-Functional Characteristics

After Functional profiles the second key set of common characteristics to profile missions and systems are the Non-Functional. The non-functional characteristics are the criteria that can be used to judge the operation of a system, such as cost or time, rather than specific behaviours (which are covered by the functional characteristics). The non-functional characteristics under consideration are applicable to both the mission (as a measure of requirements, explicit or implicit) and system (as a measure of its operation). These commonly termed 'ilities' are attributes of systems that are frequently used within the descriptions of non-functional requirements for systems. The 'ilities' have found particular acceptance in defence requirements engineering where they are often specified for military systems, e.g. survivability or manoeuvrability. For this research the 'ilities' are regarded as holistic system of systems attributes rather than being attributed to individual systems. While individual systems may be linked with or even dominate a specific attribute, it is still necessary to consider the entire system of systems contribution. This is of particular importance when dealing with emergent properties.

Various authors have identified individual 'ilities' (Rhodes 2006) and their importance for SoS engineering (Saunders, et al. 2005). Whilst the 'ilities' have gained widespread acceptance there appears to be no current universally accepted set or list of definitions for them (McManus, et al. 2007). The 'ilities' are not useable in their current form for performing an assessment of system architectures or mission requirements in general applications; instead they must be specifically understood for each application. A more structured, coherent and defined set of measurable attributes are required to characterise systems for evaluation purposes. In this research an approach has been adopted which enables these characteristics to be assessed in a structured way. This approach moves away, in most cases, from using the 'ilities' directly (since they are usually difficult or impossible to measure directly) and, instead, utilises more measurable secondary attributes and factors to characterise the attributes, as shown in Figure 26.

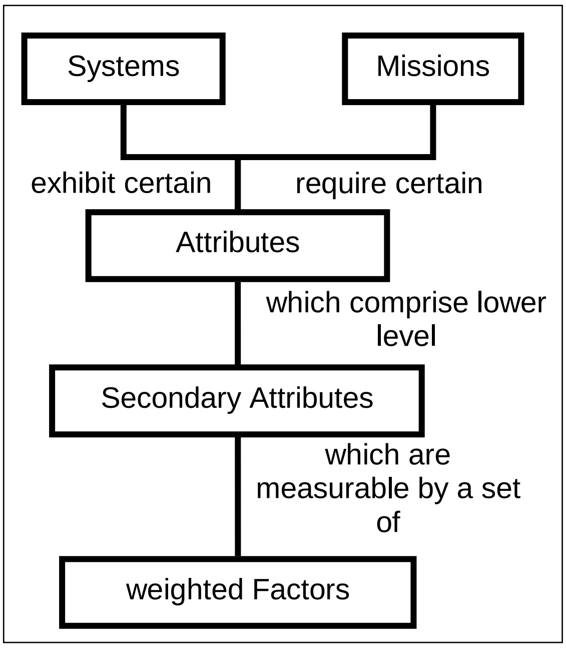


Figure 26 – Relationship of Terms

3.3.1 Attributes, Secondary Attributes and Factors

To develop list of system attributes the VPA and FFDs of the two CSAR missions presented in section 3.2 and the developed GFM were studied. Careful consideration was given to the non-functional characteristics placed on the system conducting the mission and what might influence or affect the importance of these characteristics for a particular mission. This exercise produced an initial set of attributes, secondary attributes and factors which were refined over the course of further research. Further refinement of these lists was achieved through a consideration of the relationships between Attributes, Secondary Attributes and Factors. Note that this set has been developed specifically for CSAR missions and are believed to be the most influential attributes, secondary attributes and factors for this type of mission. A universal set could be developed using this framework, allowing common understanding between mission threads and system capabilities. However, this research has bounded its focus to that of CSAR mission only and hence the sample set developed is relatively small with 12 Attributes, 29 Secondary Attributes and 48 Factors. The sets of Attributes, Secondary Attributes and Factors presented, whilst usable for CSAR missions, are not the definitive correct answer and the framework developed could be used by the systems engineering community collectively to clarify, define and evolve the naming, definitions and relationships of the characteristics. The full proposed set of 12 system attributes and descriptions as developed through the group work is shown in Table 2.

Table 2 – A Proposed Attribute Set

Attribute Name	**Attribute Description**
Adaptability	The ability of a system to change how it functions (allocation of functions) and what it does (what functions it executes) in response to the environment.
Availability	The ability to provide a particular functional level for the majority of time.
Co-operability	The ability to engage in co-operative behaviour in a team, e.g. by information sharing and mutual support.
Credibility	The impression created by a system of its intent to follow through on its actions.
Decision Making Superiority	The ability to make the right decision at the right time.
Deployability	The ability to get "on station" at the required functional level.
Effectiveness	The ability to do something effectively and properly, to deliver the required effect.
Flexibility	The allocation of functions within a system.
Interoperability	The ability to operate in synergy in the execution of assigned tasks.
Orientability	The ability to comprehend the environment.
Survivability	The ability to function during and after a natural or man-made disturbance.
Sustainability	The ability to deliver a level of performance despite any interference.

An *attribute* is defined for the purposes of this research as a high-level system characteristic. As stated previously, an attribute is not normally measurable directly. These attributes are, therefore, to be represented in terms of a set of relevant secondary attributes and, consequently, can be measured through the interpretation of appropriate secondary attributes and their contributing factors. *Secondary attributes* are normally measured through the interpretation of a subset of *factors*, but can be measured directly when they have no contributing factors. The relationship between an attribute and the relevant secondary attributes and factors is illustrated in Figure 27.

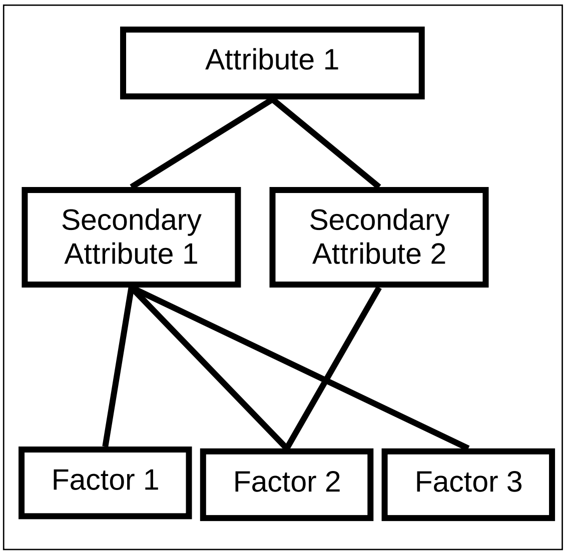


Figure 27 – Representation of an Attribute

The decomposition of attributes into secondary attributes and factors was the product of an iterative refining process. While the definitions developed were subjective a more objective approach to defining attributes, secondary attributes and factors and the relationships between them was researched. The improved method will be shown for the second version of the Decision Support System to be shown in Chapter 6. An example of the decomposition of Adaptability into one secondary attribute and its underlying factors is shown in Figure 28.

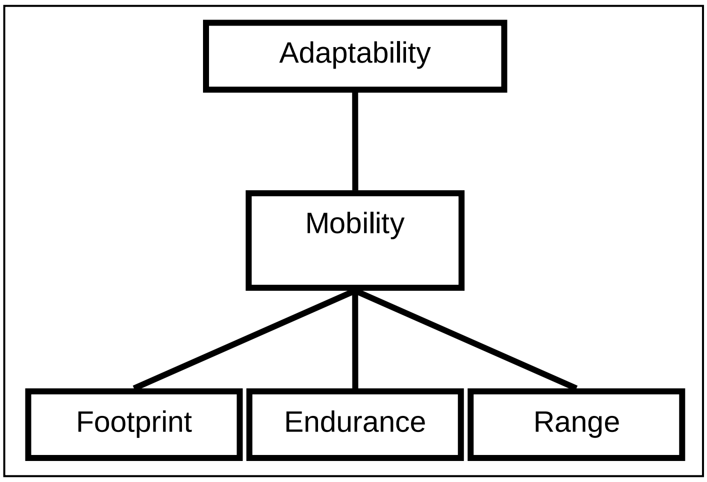


Figure 28 – Decomposition of Adaptability Attribute

An example set of proposed secondary attributes for the Effectiveness attribute is shown in Table 3, this is followed by a description of some of the measurable factors for these secondary attributes in Table 4.

Table 3 – Some Proposed Secondary Attributes for Effectiveness

Secondary Attribute Name	**Secondary Attribute Description**
Accessibility	The quality of being at hand when needed.
Configurability	The ability of a system to change its functionality through the rearrangement of its parts.
Dependability	The ability of a system to be relied upon or trusted by other systems.
Integrity	The wholeness of the system, everything within the system has a role or purpose and all inputs and outputs have somewhere to go - no internal conflicts.
Lethality	The ability to kill.
Mobility	The ability of a system to deliver an effect/output somewhere else (reach).
Predictability	The ability of a system to consistently produce the same expected results.
Timeliness	The ability to functionally achieve something within a predetermined or favourable timeframe.
Usability	How usable the system is by the operators within it (who are likely to have been trained/specialists) including recovery from errors. This does not imply simplicity.
Processing capability	The ability to process incoming knowledge, information/data effectively, efficiently and in a timely manner, to reach conclusions regardless of the state of the information/data.

Sometimes a secondary attribute is measurable itself, but more often than not it must be evaluated through the interpretation of measurable hard (technical) and soft (human/organisational) factors which are defined for the purposes of this research as low-level characteristics or properties of a system. A number of factors can be considered to contribute to a representation of a secondary attribute, as shown in Figure 27. Many of these relevant factors relate to 'soft' issues which are not necessarily objectively measurable but for which rational subjective judgement can be applied using, for example, the Soft Factors Modelling Tool (SFMT) that was developed in the overall SEAS DTC research project that this thesis contributes to (Hodgson and Siemieniuch 2009). The SFMT focuses on the impact of soft factors relating to cultural values on communicating and implementing decisions and is described in more detail in Siemieniuch and Sinclair (2006). It should be emphasised that the DSS approach does not differentiate 'soft' from 'hard' issues – rather it recognises that the various attributes can be affected by factors from across the whole soft/hard spectrum. A small sample of proposed factors is shown below in Table 4 for the Mobility secondary attribute. Each factor can contribute, of course, to the evaluation of a number of secondary attributes (as shown previously in Figure 27).

Table 4 – Some Proposed Factors

Factor Name	**Factor Description**
Footprint	The physical size and weight of the system.
Endurance	The time and distance a system can operate for on its own resources.
Range	The range at which the system can deliver effects/outputs.

The full set of developed Attributes, Secondary Attributes and Factors are presented in Appendix G.

3.3.2 Architecting Non-Functional Characteristics into an Approach for a DSS

With a set of Attributes, Secondary Attributes and Factors developed which can be used to profile both missions and available SoS a consideration was given as to how it could be utilised in the overall concept of the architecture of an approach for a DSS as outlined previously in Figure 25. An architecture utilising the second component identified (the developed set of attributes, secondary attributes and factors) was created as illustrated in Figure 29.

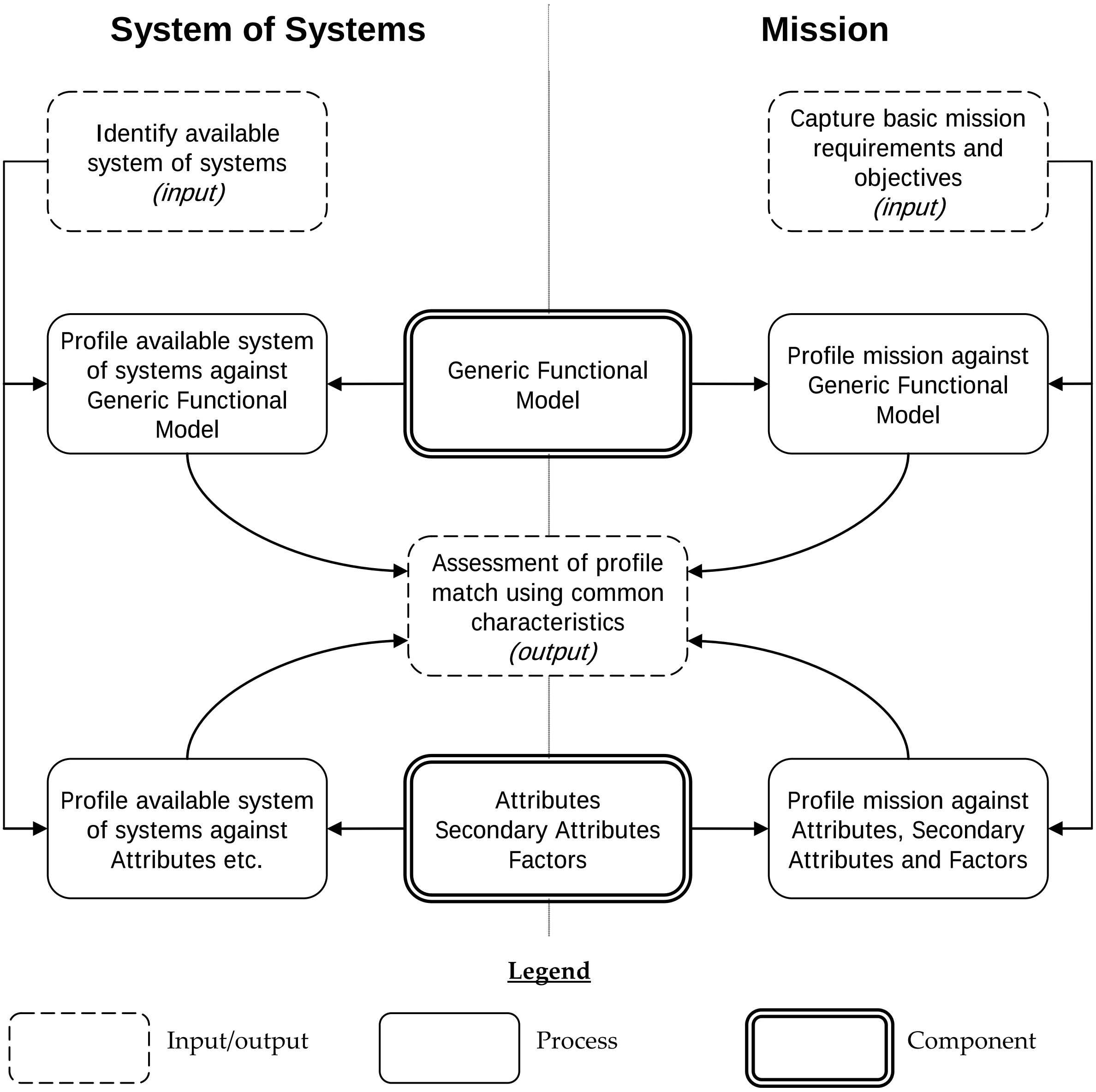


Figure 29 – An Architecture of an Approach for a DSS Utilising Attributes

With a common set of non-functional characteristics captured in the developed set of attributes, secondary attributes and factors the next stage was to consider the physical environment characteristics.

3.4 Physical Environment Characteristics

The third key set of characteristics to profile missions and systems is the Physical Environment. The physical environment captures characteristics of the environment (climate, terrain, political, hostiles, etc.) in which the mission will be conducted. This can be used as a first pass filter to ensure that system options are appropriate for the mission's conditions, helping to reduce the number of systems options under consideration. Most military assets have specified operating conditions under which they will function (temperatures, etc.) and the Physical Environment captures this knowledge and allows it to be compared to the mission's physical environment requirements.

Table 5 – Overview of the Physical Environment Characteristics

Characteristic	Description
Climate	The type of climate the mission will be conducted in as according to the Köppen climate classification (Kottek, et al. 2006).
Environment	The type of operating environment the mission will be conducted in, e.g. Land, Sea, Air, etc.
Predictability	An indication of how likely the situation is to change over the course of the mission.
Strategic Purpose	The overall strategic purpose of the mission, e.g. offensive, defensive, operations other than war.
Political Environment	An indication of the hostility of the local populace to friendly forces and the political stability of the area where the mission will be conducted.

The full set of physical environment characteristics are shown in Appendix H.

3.4.1 Architecting Physical Environment Characteristics into an Approach for a DSS

With a set of physical environment characteristics developed which can be used to profile both missions and available SoS a consideration was given as to how it could be utilised in the overall concept of the architecture of an approach for a DSS as outlined previously in Figure 29. An architecture utilising the third component identified (the developed set of physical environment characteristics) was created as illustrated in Figure 30.

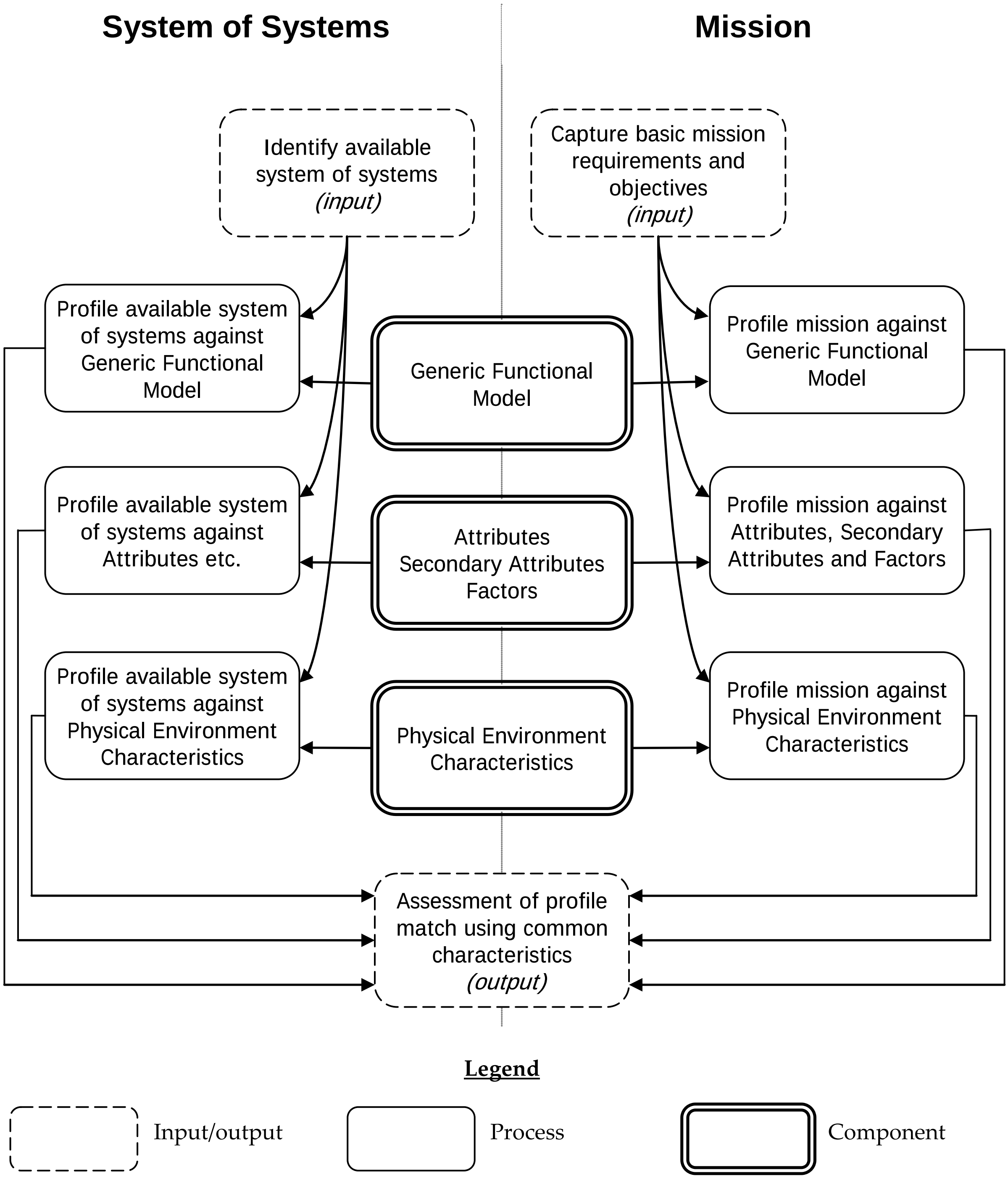


Figure 30 – An Architecture of an Approach for a DSS Utilising Common Characteristics

Three main characteristics of systems of systems and missions have been shown, the first of which is the matching of functional characteristics through the use of a developed GFM. The second is the non-functional or "ilities", which have been captured through a proposed hierarchical set of attributes, secondary attributes and factors. Finally the third set of characteristics shown was a consideration of the physical environment. The next section describes the development of an architecture of an approach to develop a decision support system that utilises these sets of characteristics to support the decision maker in the choice of which system to use for a particular CSAR mission.

3.5 An Architecture of an Approach for a Decision Support System

The Decision Support System (DSS) methodology was developed by defining a generic functional model for CSAR missions as reported in the previous section: this established the generic functional characteristics that a CSAR mission is likely to require. Separately the non-functional characteristics were considered and a generic set of attributes, secondary attributes and factors was devised. A simple method was envisioned to allow these generic sets to be re-profiled to a particular mission by weighting the generic set elements which would allow unimportant characteristics to be disregarded by a zero or *not applicable* weighting. This will be discussed in the implementation of the approach in the next Chapter 4.

At any time, some given resources or systems are made available to carry out a particular CSAR mission. These systems (human and technical) may or may not have the inherent skills and competencies needed to carry out the specific mission. For a given CSAR mission various combinations of systems can be brought together to form a SoS to carry out that mission. In order to assess the suitability of the available SoS alternatives, they can be assessed in terms of the same characteristics as used for the mission. These characteristics may be given priority weightings to recognise their relative importance within specific missions. The basic mission requirements and objectives are captured and used to profile the mission in terms of required functional, non-functional and physical environment requirements. The mission's functional characteristics are ascertained by weighting the developed generic functional model in terms of the mission requirements and objectives. The mission's non-functional characteristics are captured by weighting the developed set of attributes, secondary attributes and factors in terms of the characteristics needed to achieve the mission requirements and objectives. The physical environment characteristics are identified by weighting the developed set of physical environment characteristics.

Available SoS which could be used are profiled against the generic functional model, the set of attributes, secondary attributes and factors and the physical environment characteristics. The available SoS are then simply matched to the mission in terms of functional, non-functional and physical environment characteristics. The functional and non-functional perspectives of the potential system of systems provide a capability overview of each to the decision makers, allowing them to compare the alternative system of systems against the mission needs.

The developed approach is compatible with the Recognition Primed Decision Model identified previously in section 2.4, shown again in Figure 31. It supports the *situation* by allowing the decision maker to quickly profile the mission against generic models of that mission helping to identify oversights and errors. The main support is given to the *will it work?* question as the actions, expressed as SoS, can be directly assessed against the situation (mission).

Figure 31 – Recognition Primed Decision Model (Klein, Calderwood and MacGregor 1989, 464)

The developed approach can be used from either an open or closed viewpoint. The viewpoint taken is dependent on the method of implementation and this will be discussed in the next Chapter 4. An overview of the architecture of an approach to developing a DSS is shown in Figure 32 with the components developed in this chapter shaded.

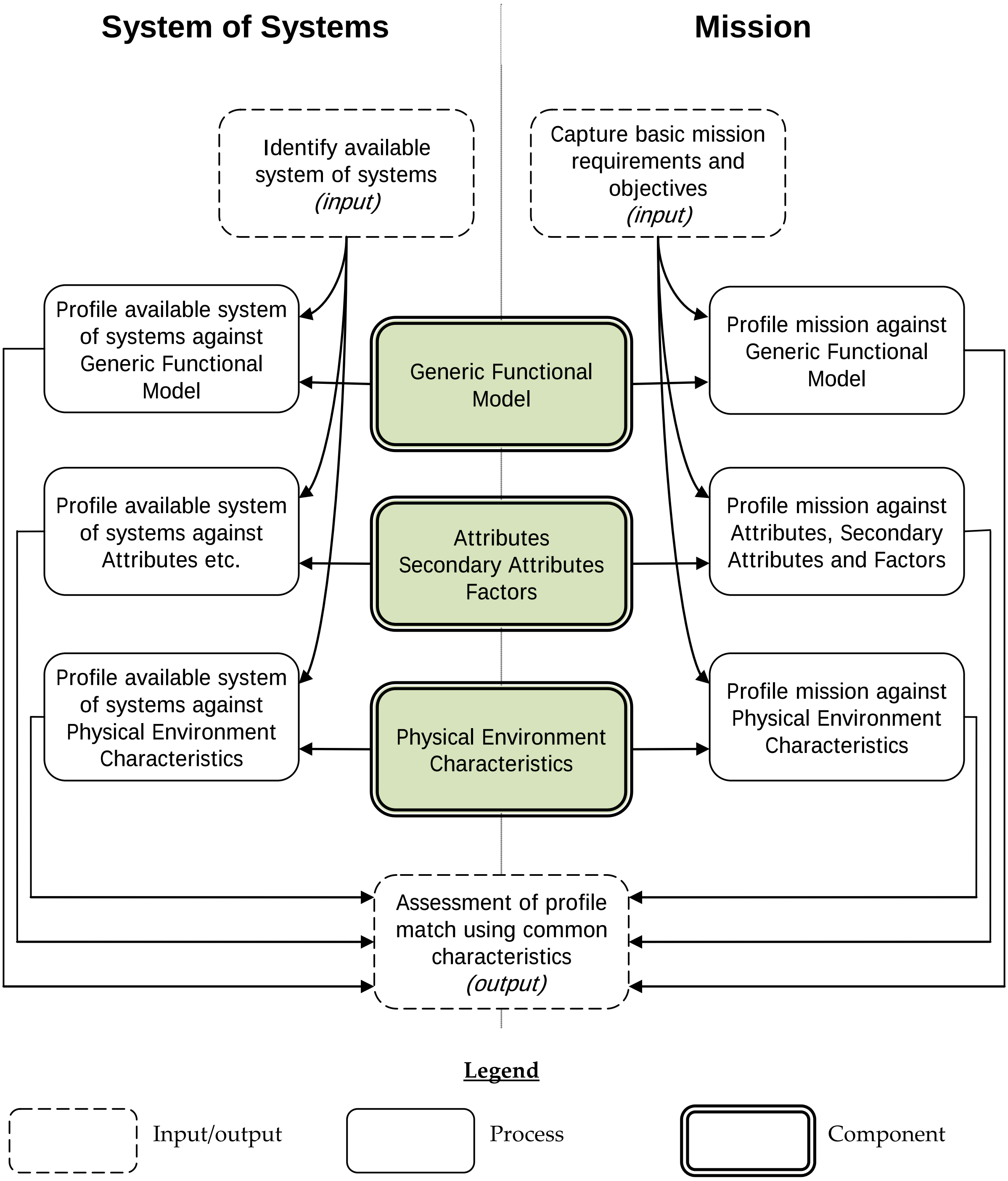


Figure 32 – An Architecture of an Approach for a DSS Utilising Environment Characteristics

Note that the assessment produced by the approach shown in Figure 32 may indicate that the available systems cannot be combined to complete the mission. Whilst this evaluation process will not give guarantee of success it will show whether a system is functionally and non-functionally suited to the characteristics of the mission and capable of operating in the mission's physical environment. It is likely that, during a specific mission, circumstances may change or objectives may be altered. The outlined approach can be used at any stage of the mission, from pre-mission planning to mission completion and post-mission evaluation, to assess the available SoS and identify alternative SoS to accommodate changes in circumstances or mission objectives.

3.6 Chapter Summary

This chapter has described the development of an architecture of an approach for a decision support system. This approach utilised common sets of characteristics to profile missions and available SoS to create comparative profiles to support the decision maker. Three key common sets of characteristics were identified for a decision support system: functional, non-functional and physical environment. These common sets were developed as no existing sets were identified. The functional characteristics were captured using Viewpoint Analysis and Functional Flow Diagram techniques to develop a Generic Functional Model which captures the functionality required to conduct the majority of combat search and rescue missions. The non-functional characteristics were identified in the literature as the *ilities* but no current universally accepted set of list of definitions could be found for them. The *ilities* were also unusable in their current form as they are generally defined for specific applications as opposed to the more abstract and commonly usable definitions sought for this research. They are also difficult to measure directly due to their scope. To address these issues a hierarchical set of Attributes, Secondary Attributes and Factors were defined which can be used to profile both missions and available SoS. The Physical Environment characteristics were identified as important as the systems considered to conduct a mission must be capable of operating in the mission's environment. A broad set of physical environment characteristics were identified to fulfil this need.

With three common sets of characteristics identified an architecture of an approach for a decision support system was presented that can support the RPD model of decision making identified in Chapter 3 as the most representative model of how military decision makers actually make decisions. The next chapter will consider the implementation of this architecture of an approach for a decision support system.

Chapter 4

Implementing an Approach for a Decision Support System

Outline of Chapter

This chapter outlines the prototype tool developed as an implementation of the research presented in the previous Chapter 3. The implementation is discussed and how the missions and systems are profiled is detailed. With the profiles established the method of assessment of SoS profiles against mission profiles is presented. A set of case studies is conducted using the developed decision support system and the results are reported. Finally the case studies are evaluated and the approach reassessed in light of these findings.

Research Contributions of Chapter

From the previous chapter(s):

*This **Research** established a **Problem** and **Objectives** to develop an **Architecture** of a **Parametric Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics**. This set of common characteristics which includes **Functional** and **Non-Functional** characteristics, was used to inform the decision maker. The non-functional characteristics were characterised by **Attributes** which are composed of **Secondary Attributes** which, in turn, are composed of **Factors**. The functional characteristics were captured in a **Generic Functional Model** for combat search and rescue missions.*

This chapter introduces:

*The architecture has been implemented in a **Software Tool**. The software tool was used to conduct a set of literature based **Case Studies** of the application of this research (combat search and rescue) and the results evaluated.*

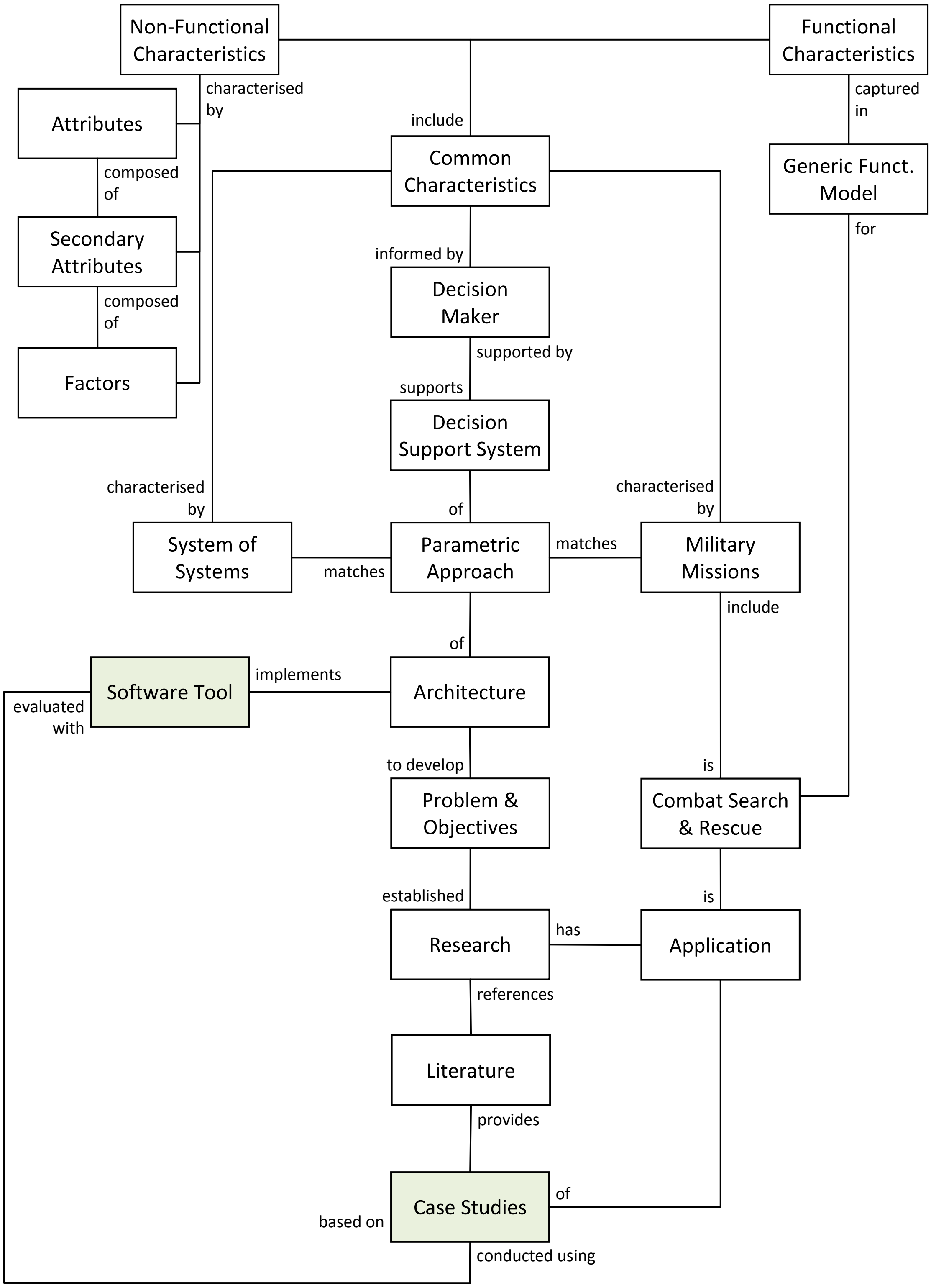


Figure 33 – Logical Model of Research Contributions of Chapter Four

4.1 Implementation Introduction

As stated in previous chapters, the purpose of this research is to move beyond rigidly structured decision making processes that are isolated from the environment towards a dynamic, informed and usable approach that aids decision makers and allows system knowledge to be cumulatively acquired through use. The tool is a decision aid, it does not dictate to the decision maker which SoS to use nor does it indicate any likelihood of mission success. The tool advises the decision maker of potential problems with the SoS in terms of the mission's demands that they may have not identified. This decision aid addresses two main problems faced by decision makers. First the tool addresses any lack of understanding of the systems in terms of configuration or capability changes. Secondly the tool provides an open viewpoint of the problem which negates the closed viewpoint of the overall Recognition Primed Decision (RPD) model. The RPD was introduced in section 2.4 as the most representational model of how military decision makers actually make decisions. In the RPD model the decision maker chooses a solution based on previous experience which is a closed viewpoint. Hence the open viewpoint that this implementation aims to provide will provide a second opinion from another viewpoint to the decision maker. An initial architecture of an approach for a DSS utilising commons sets of functional and non-functional characteristics was developed in the previous Chapter 3. This chapter aims to realise this architecture in a prototype software tool.

Three main aims were set to be achieved by developing this first prototype software tool:

1. **Feasibility** - demonstrate that the developed DSS architecture could be implemented in a software tool.
2. **Validation** - perform a set of historical case studies utilising the developed prototype software tool.
3. **Guide** - guide the further development of the architecture of an approach for a DSS.

We start by considering the architectural implications of implementing the developed approach.

4.2 Developing a Prototype Decision Support System Software Tool

The developed architecture for the DSS is shown in Figure 34. This considers only the SoS, not how they are assembled or the component systems. From an implementation perspective the developed architecture presents a problem in that it requires the SoS to be considered as a whole to be profiled by the developed sets of common characteristics. Considering the whole allows an open viewpoint to be taken on the SoS, but in practical terms this is difficult to realise as the developed sets of characteristics must be assessed for each SoS individually. If the characteristics could be assessed at the system level and then combined to the chosen SoS configuration then only the systems would need to be profiled, not the more numerous potential SoS combinations.

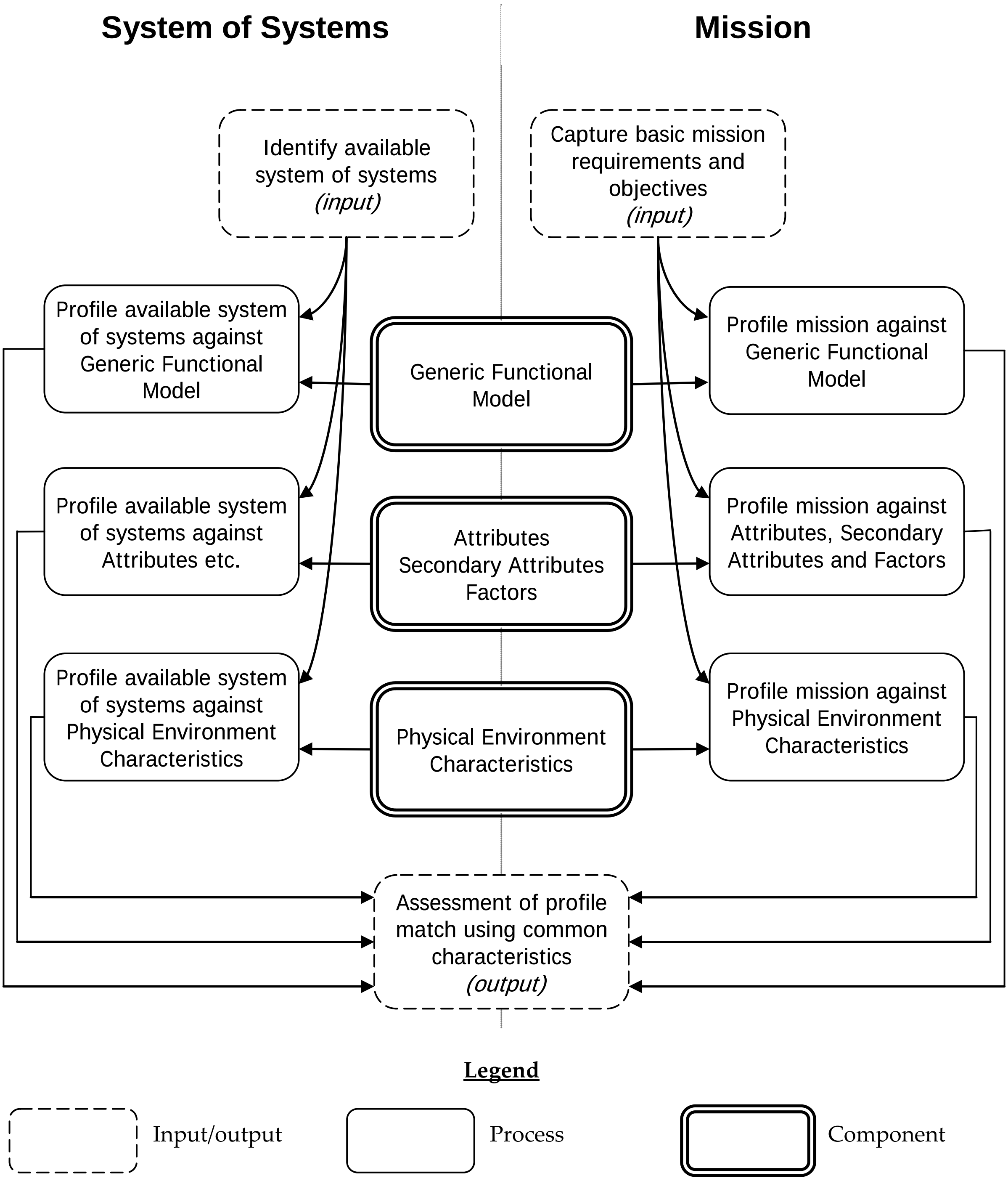


Figure 34 – An Architecture of an Approach for a DSS: Profiling the SoS

4.2.1 Shifting from the SoS to the Systems Viewpoint

It was recognised that it would be much faster and more practical to assess systems individually and then somehow combine their measurements into SoS options. This has the advantage that only systems need to be characterised, not the more complex whole SoS. If a satisfactory way were found to combine system measures into SoS it would also allow all possible SoS combinations to be considered from a base set of available systems, as illustrated in Figure 35.

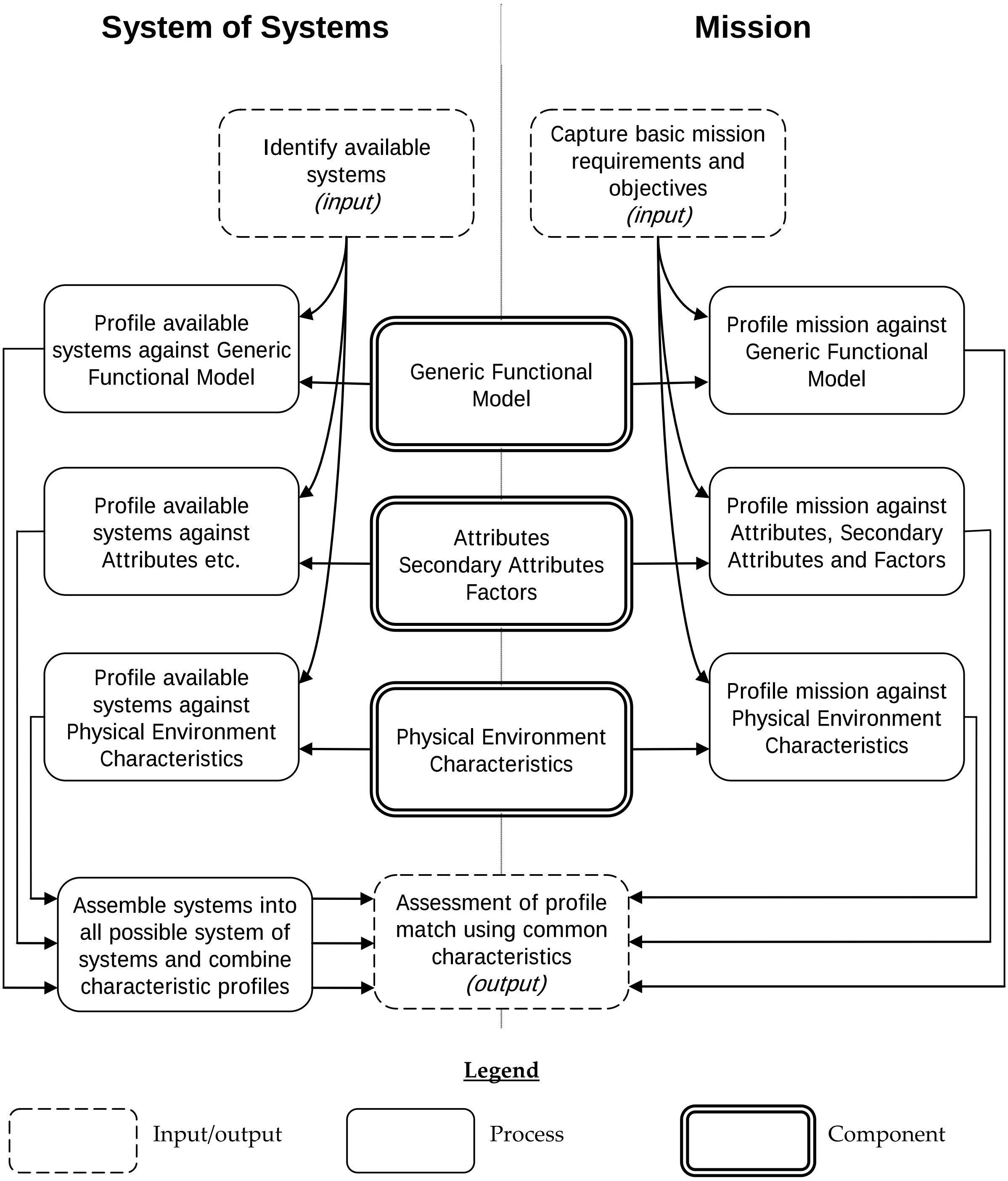


Figure 35 – An Architecture of an Approach for a DSS: Profiling Systems

This architecture shown in Figure 35 is a closed reductionist approach as opposed to the needed open viewpoint required by the research. The individual assessment of systems leading to a holistic SoS is a closed reductionist approach as it considers each system in isolation and then amalgamates the combined measures via a simple mechanism. This mechanism simply checks that at least one of the component systems can achieve the required mission characteristic. This mechanism is detailed in subsection 4.2.4. The open properties of the system are not considered such that the resultant SoS is the sum of the whole, no less and no greater. However, as a starting point to consider a decision support system this will suffice to set the stage for later development work which can address a more open approach. This first prototype software tool will develop the characterisation approach to match missions and systems using common metrics. The feasibility can be tested using a closed viewpoint DSS. Once the feasibility of this approach has been established approaches can be investigated to allow both open and closed viewpoints on the metrics established in the initial prototype software tool and tested against the case studies previously utilised.

4.2.2 Prototype Software Tool Implementation

The prototype software tool was required to implement the architecture shown in Figure 35 and allow profiling of a mission and up to four systems against the developed sets of common functional, non-functional and physical environment characteristics. The prototype software would also need to automate the collection of systems into possible SoS combinations and generate the combined characteristics for each.

As the prototype software tool would need to handle a lot of tabular data (as the common characteristics were expressed in this way) the Microsoft® Excel® spreadsheet software package was selected to implement the tool. The developed DSS provides a structured software interface to input, record and compare profiles of system of systems options against mission requirements. The following process is followed by the user to use the prototype software tool:

- A cover sheet is completed detailing the mission and outlining the actual sequence of events if a historical case study.
- Mission requirements are profiled against a standard set of Functional, Non-Functional and Physical Environment metrics
- Mission requirements are measured against the profiled set of Functional, Non-Functional and Physical Environment metrics

- Measurement is made of up to four systems (a limitation of the prototype tool's implementation) against profiled Functional, Non-Functional and Physical Environment characteristics.

Once the user has completed the following steps above the prototype software tool automatically:

- Indicates a simple pass/fail status for each characteristic determined by mismatch between mission and system profiles
- Generates all possible SoS combinations and highlighting of failures against mission requirements

In the prototype software tool the mission/asset measurements are done on separate worksheets in Microsoft® Excel® and a final worksheet presents an overview of all the system characteristics and all of the system of systems pass/fail indicators in a matrix layout. The decision maker can then compare the two overviews and hence assess SoS against the particular mission. Throughout the workflow the developed prototype tool highlights mismatches between the mission requirements and the assets/systems and it is left to the decision maker to assess the impact of each mismatch. This approach is in concordance with Rittle and Webbers (1973) definition of wicked problems in that "solutions to wicked problems are not true-or-false, but good-or-bad" (see Appendix A for the full definition of wicked problems). The next subsection considers the profiling of the mission and systems in more detail.

4.2.3 Profiling Missions and Systems in the Prototype Software Tool

In the context of this research a profile is specifically defined as *a representation of the structured set of common characteristics of a mission or a system of systems*, as previously defined in section 3.1. Profiling of missions and systems in the prototype software tool is achieved by the user customising the value of the characteristics for each mission to a simple comparative such as true/false or high/low as appropriate. Unimportant characteristics are removed from consideration by the user setting their value to *n/a* (not applicable), which will grey out the characteristic across the systems and system of systems. These measures are greyed out as opposed to hidden as they may be used by the decision maker if two systems are equally suitable to conduct the mission according to the other characteristics. The mission is profiled first and then the available systems. As the mission has been profiled first feedback can be given to the system profile as to how it matches the mission's profile. This is achieved through conjunction of the two profile measures of a particular characteristic (a logical AND). Note that the mission may have negatives against certain characteristics as the characteristic sets developed are considered broad enough to cover the majority of CSAR missions, hence it's unlikely that a particular CSAR mission will require *all* of the characteristics. If the mission's profile has a negative against a function and the system does as well then that is a *pass* as the two values match. This can be seen in an example of a functional profile input shown in Table 6 for a *MH-53J 'Pave Low' Helicopter,* one of the systems profiled for case study two later in section 4.3.2.

Table 6 – Example of a Profile of Systems and Mission Against GFM

	Name	**MH-53J 'Pave Low' Helicopter**			
	Basic Configuration	Huey Helicopter, Pilot, Co-Pilot/Navigator, Crew Chief, Medic			
High Level GFM Function ID (1)	High Level GFM Function Name (1)	Low Level GFM Function ID (2)	Low Level GFM Function Name (2)	System Rating	Mission Match?
1	Detect distress call	1.1	Receive distress call	N	fail
		1.2	Collate distress information	N	fail
		1.3	Validate distress information	N	fail
		1.4	Acknowledge receipt	N	fail
2	Collaborate with coalition	2.1	Vet information for security	N	pass
		2.2	Transfer knowledge	N	pass
		2.3	Negotiate assistance	N	pass
3	Planning & Adapting	3.1	Assess situation	Y	pass
		3.2	Plan rescue	Y	pass
		3.3	Select appropriate resources	N	fail
		3.4	Prepare resources	N	pass
		3.5	Conform to the laws of war	Y	pass
4	Execute Action	4.1	Navigate	Y	pass
		4.2	Ingress	Y	pass
		4.3	Locate rescue target	Y	pass
		4.4	Deal with hostile forces	N	fail
		4.5	Egress	Y	pass
		4.6	Deal with civilians	Y	pass
		4.7	Deal with rescue target	Y	pass
		4.8	Sustain assets	Y	pass
5	Command	5.1	Gather information	Y	pass
		5.2	Monitor progress	Y	pass
		5.3	Make decision	Y	pass
		5.4	Disseminate information	Y	pass
6	Assess Mission	6.1	Debrief rescued asset	N	pass
		6.2	Debrief team	N	pass
		6.3	Evaluate mission performance	N	pass
		6.4	Generate lessons learnt	N	pass
		6.5	Recommend doctrine update	N	pass

A similar input panel is completed for each system and for the non-functional (attributes, secondary attributes and factors) and physical environment characteristics as well, as can be seen in the following section 4.3. This approach allows a *catalogue* of systems to be created which can be quickly pulled into SoS configurations without needing to re-profile them in terms of the reduced set of characteristics that the mission is being measured with.

4.2.4 Method of Assessment of SoS Profiles Against Mission Profiles

With the method of inputting data to the tool described in the previous section 4.2.3 this section describes the assessment of SoS profiles against mission profiles. The *assessment* was envisioned to be a comparative assessment allowing a proposed SoS Profile to be compared to the mission's profiles and mismatches highlighted between the two as described previously in section 3.1. In the developed approach the assessment of a SoS to conduct a particular mission is driven by sets of common characteristics that are unrelated parameters. This section describes the realisation of this process. To start with we consider the assessment of individual systems.

The systems profile output is shown in Table 7. The table shows two FFD levels of functions from the GFM. The mission is profiled against each of these functions. The requirement of the function in order to complete the particular mission is indicated by a simple Y/N (Yes/No). If the function is not required the measure is greyed as it is not required by the mission and hence the available systems ability or inability to fulfil that function is not considered important. If the function is required the box is highlighted in green to emphasise its requirement and aid comparison against the individual systems. The individual systems are each profiled in the next columns starting with the *MH-53J 'Pave Low' Helicopter*. The profile is directly imported from the inputs as shown previously in Table 6. If the system can perform the function then it is highlighted in green, if not in red to aid comparative evaluation by the decision maker. Note that the functional profile is shown for illustrative purposes and additional profiles would be completed for the non-functional and physical environment characteristic sets.

To enable the assessment of SoS profiles against a mission profile the individual system profiles must be combined into SoS profiles. The formation of SoS from individual systems is achieved through logical disjunction (OR) for each characteristic. Hence, if one of the component systems of an SoS achieves the characteristic state required by the mission then the SoS is considered to achieve that characteristic (even if the other component systems do not). The developed software tool is therefore a framework and algorithm for SoS generation from a set of available systems.

Table 7 - Example of a Summary Profile of Systems and Mission Against the GFM

Function ID (1)	Function Name (1)	Function ID (2)	Function Name (2)	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)
1	Detect distress call	1.1	Receive distress call	Y	N	N	Y	N
		1.2	Collate distress information	Y	N	N	Y	Y
		1.3	Validate distress information	Y	N	N	Y	Y
		1.4	Acknowledge receipt	Y	N	N	Y	N
2	Collaborate with coalition	2.1	Vet information for security	N	N	N	N	Y
		2.2	Transfer knowledge	N	N	N	N	Y
		2.3	Negotiate assistance	N	N	N	N	Y
3	Planning & Adapting	3.1	Assess situation	Y	Y	Y	Y	Y
		3.2	Plan rescue	Y	Y	N	N	Y
		3.3	Select appropriate resources	Y	N	N	N	Y
		3.4	Prepare resources	N	N	N	N	N
		3.5	Conform to the laws of war	Y	Y	Y	Y	Y
4	Execute Action	4.1	Navigate	Y	Y	Y	Y	N
		4.2	Ingress	Y	Y	Y	Y	N
		4.3	Locate rescue target	Y	Y	Y	Y	N
		4.4	Deal with hostile forces	Y	N	Y	Y	N
		4.5	Egress	Y	Y	Y	Y	N
		4.6	Deal with civilians	Y	Y	N	N	N
		4.7	Deal with rescue target	Y	Y	N	N	N
		4.8	Sustain assets	Y	Y	N	N	N
5	Command	5.1	Gather information	Y	Y	Y	Y	Y
		5.2	Monitor progress	Y	Y	N	Y	Y
		5.3	Make decision	Y	Y	Y	Y	Y
		5.4	Disseminate information	Y	Y	Y	Y	Y
6	Assess Mission	6.1	Debrief rescued asset	N	N	N	N	N
		6.2	Debrief team	N	N	N	N	N
		6.3	Evaluate mission performance	N	N	N	N	N
		6.4	Generate lessons learnt	N	N	N	N	N
		6.5	Recommend doctrine update	N	N	N	N	N

To consider if an SoS is suitable the three sets of characteristics (functional, non-functional and physical environment) are combined using logical conjunction (AND). Hence if the SoS matches the missions requirements for each characteristic then it is deemed suitable. An example of the SoS profile output of the tool is shown in Table 8. This shows only the functional profile for illustration proposes, additional profiles for the non-functional and physical environment characteristics would also be completed, as can be seen in the case studies in the following section 4.3.

Table 8 - Example of a Summary Profile of SoS and Mission Against the GFM

Function ID (1)	Function Name (1)	Function ID (2)	Function Name (2)	Mission	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Detect distress call	1.1	Receive distress call	Y	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
		1.2	Collate distress information	Y	FAIL	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.3	Validate distress information	Y	FAIL	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.4	Acknowledge receipt	Y	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
2	Collaborate with coalition	2.1	Vet information for security	N	N	N	N	N	N	N	N	N	N	N	N
		2.2	Transfer knowledge	N	N	N	N	N	N	N	N	N	N	N	N
		2.3	Negotiate assistance	N	N	N	N	N	N	N	N	N	N	N	N
3	Planning & Adapting	3.1	Assess situation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.2	Plan rescue	Y	Y	Y	Y	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.3	Select appropriate resources	Y	FAIL	FAIL	FAIL	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.4	Prepare resources	N	N	N	N	N	N	N	N	N	N	N	N
		3.5	Conform to the laws of war	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
4	Execute Action	4.1	Navigate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.2	Ingress	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.3	Locate rescue target	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.4	Deal with hostile forces	Y	Y	Y	Y	Y	Y	Y	Y	Y	FAIL	Y	Y
		4.5	Egress	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.6	Deal with civilians	Y	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
		4.7	Deal with rescue target	Y	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
		4.8	Sustain assets	Y	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
5	Command	5.1	Gather information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.2	Monitor progress	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.3	Make decision	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.4	Disseminate information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
6	Assess Mission	6.1	Debrief rescued asset	N	N	N	N	N	N	N	N	N	N	N	N
		6.2	Debrief team	N	N	N	N	N	N	N	N	N	N	N	N
		6.3	Evaluate mission performance	N	N	N	N	N	N	N	N	N	N	N	N
		6.4	Generate lessons learnt	N	N	N	N	N	N	N	N	N	N	N	N
		6.5	Recommend doctrine update	N	N	N	N	N	N	N	N	N	N	N	N

The next section considers two historical case studies conducted with the developed software tool.

4.3 Case Studies

The developed architecture of an approach for a DSS is shown again in Figure 36. The last few subsections have detailed the implementation of this architecture in a software tool. This section discusses two historical case studies which were conducted using the developed software tool. Each case study followed the approach outlined in Figure 36 with the mission and available systems profiled against the developed standard sets of Functional, Non-Functional and Physical Environment characteristics. The final output was an assessment of the mission and SoS profile match. This assessment will showed the suitability or otherwise of the available SoS configurations. Comparing this suitability to the actual SoS used in the mission helps to validate the utility of this tool in addressing the overall research problem.

Conducting the case studies with the developed software tool will demonstrate the feasibility of the approach and the success of its implementation in a software tool. Each case study will demonstrate the validity of the approach and serve as a guide to future development within this research. In addition the case studies will seek to demonstrate the profiling of missions and systems using common sets of characteristics, the formation of SoS profiles from the system profiles, the matching of mission and SoS profiles and the overall utility and value of the output of this exercise to a military decision maker. These will be addressed in the summary of the case studies in subsection 4.4.

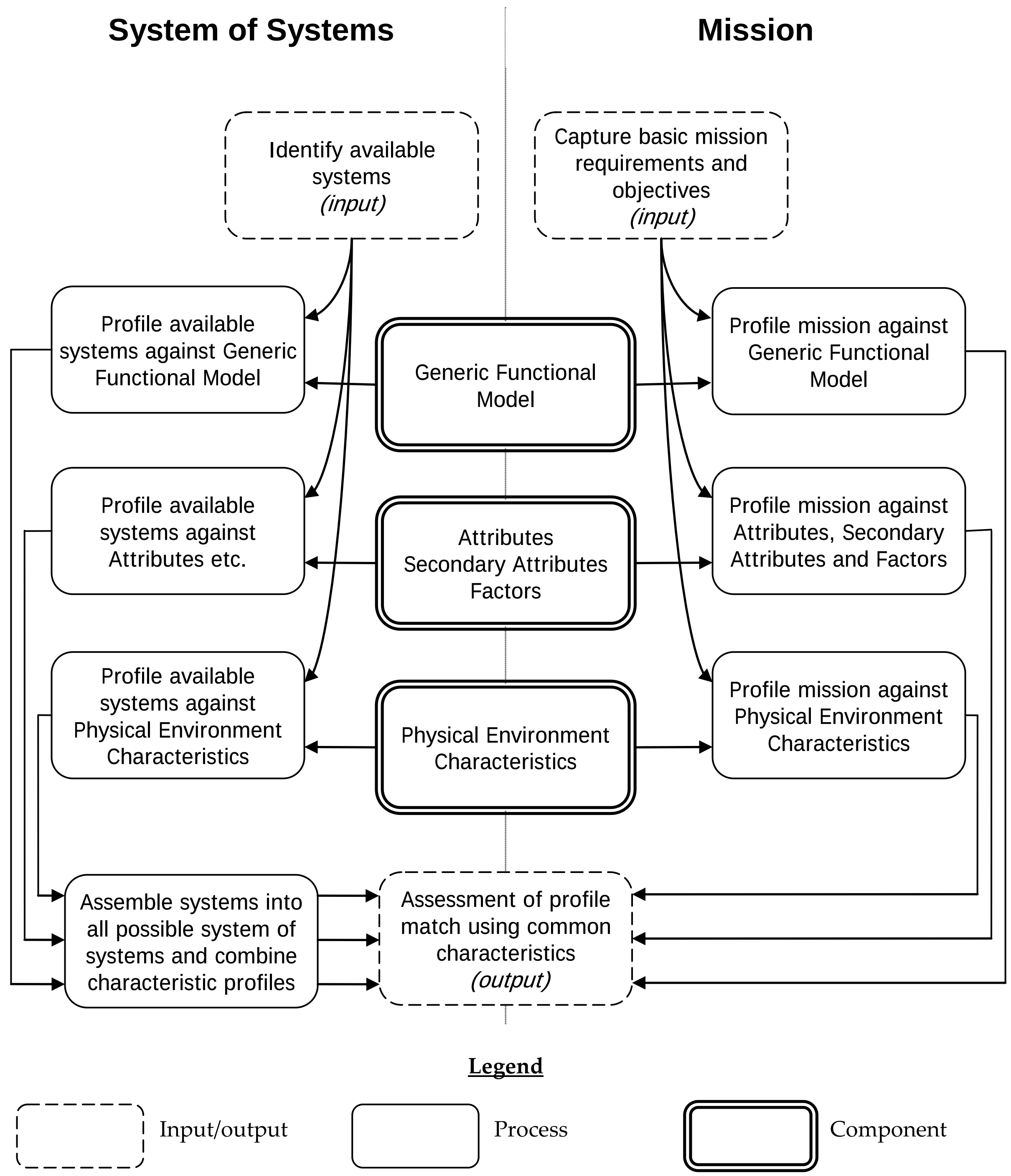


Figure 36 – Implemented Architecture of an Approach for a DSS

Two CSAR case studies were chosen for analysis, which were based on eyewitness accounts and statements. The first case study follows the 1971 rescue of a downed Navy Seawolf helicopter crew near Tra Vinh, Vietnam, during the Vietnam War (Evans 1999). The second case study concerned the 1991 rescue of Lt Devon Jones, an F-14 pilot downed near Baghdad, during the Gulf War (Evans 1999). For each case study the general background to the mission was noted, the sequence of key events was established and the key systems used identified. The systems used in the mission were profiled and compared to the mission's requirements. The systems were compiled into all the possible system of systems combinations and each combination was assessed against the mission's requirements using the implemented approach shown previously in Figure 35. The following two subsections 4.3.1 and 4.3.2 discuss case study one and two respectively.

4.3.1 Case Study One – Seawolf CSAR in Vietnam

This subsection is supported by Appendix I and considers a historical CSAR rescue based in the Tra Vinh area of Vietnam, during the Vietnam War (Evans 1999, 75), a brief synopsis of which follows.

4.3.1.1 Case Study One Narrative

On the 3rd December 1971 a Navy Seawolf helicopter was hit by enemy fire and went down near Tra Vinh. A mayday call from this helicopter was received by 'Paddy Control', the US Air Force controllers who provided the majority of radar coverage for the Delta. Paddy Control contacted 'Dustoff Operations', the operational command of US aero-medical evacuation missions in Vietnam, with an emergency CSAR mission. Captain David Freeman, a Dustoff pilot from the 57th Medical Detachment, scrambled immediately. It took 15 minutes for Captain Freeman to reach the last known radar coordinates of the Seawolf and he was vectored there by Paddy Control. Not knowing the status of the Seawolf's crew and the tactical situation on the ground Captain Freeman flew a search pattern and the crew performed a visual search. After a couple of minutes searching the crew chief located the wreckage of the Seawolf which had crashed in a large open area. This was a tactically bad situation for the crew as they were easily visible and had little cover. Four airmen were identified using their personal weapons to defend the crashed Seawolf's position from a squad of Vietcong. As Freeman passed overhead of the crash site he received a call on his Ultra High Frequency (UHF) radio from another Seawolf which was en route to their location. The Seawolf gave his Estimated Time of Arrival (ETA) as 5 minutes but, upon hearing Freemans explanation of the situation on the ground, made it 3 minutes. The best course of action had been decided upon by the Dustoff crew and the Dustoff landed behind the downed Seawolf which would act as cover. As the downed Seawolf crew broke cover and ran to the waiting Dustoff the supporting Seawolf arrived on station and engaged the Vietcong with its arsenal of guns and rockets. As the Vietcong went to ground the downed Seawolf crew boarded the Dustoff which exited unscathed from the way it had approached. (Evans 1999, 75-76)

This historical rescue has been chosen due to the highly reactive, dynamic nature of the rescue operation. Taking the Dustoff Huey helicopter working with Dustoff Operations and Paddy Control as the starting SoS for the mission it quickly evolves to incorporate the flying Seawolf that provided the much needed support at the end of the mission. This case study was conducted using the implemented approach described previously in this chapter and sought to understand the implications of the various SoS configurations against mission requirements. The step by step events of the historical case study are presented in Appendix I.1, Sequence of Events.

4.3.1.2 Case Study One Functional Profile

The full set of functional profiles are presented in Appendix I.2. The functional profiles have been taken from the developed GFM. The four systems profiled for this case study are:

- A rescue configuration Huey helicopter (functionally profiled in Appendix I.2.1)
- An attack configuration Huey helicopter (functionally profiled in Appendix I.2.2)
- Air Force Controller (functionally profiled in Appendix I.2.3)
- 57th Medical Attachment Controllers (functionally profiled in Appendix I.2.4)

These systems were profiled by the researcher based on available information. A subject matter expert was sought to aid with this process but none were available through the SEAS DTC. Note that due to the specialised nature of this research the number of suitably qualified subject matter experts is very limited and hence approaching the SEAS DTC was the only available potential point of access to them.

The profiled systems were combined into all possible SoS configurations, as shown in the functional matrix against the missions functional profile in Appendix I.2.5, Functional Profile – Forming Systems. The systems were combined simplistically by indicating a 'pass' against the various metrics if at least one of the component systems passed. As described previously in section 4.2.3 no consideration was given in the tool to the cumulative effect of multiple systems with different attributes.

The various SoS configurations assessed showed that only the combination of the rescue Huey (Dustoff), the attack Huey (Seawolf), the Air Force Controllers (Paddy Control) and the 57th Medical Attachment Controller (Dustoff Operations) provided all of the functionality that the mission required. Whilst simplistic the pass/fail indication of the assessment gave a clear view of the potential functionality gaps of a subset of this SoS. It is left to the decision maker to determine the impact of the missing functionality for the missions. For example, if a critical function such as *4.7 deal with rescue target* was absent (which would prevent the SoS from actually rescuing the downed personnel) this would be an unacceptable SoS to use for the mission. However, if an alternative had for example the function *1.4 acknowledge receipt* absent from its profile the decision maker may decide to still go with this SoS over the previous example. Down selection could be improved by applying filters to critical functionality and hence discounting those SoS configurations which were missing them.

4.3.1.3 Case Study One Non-Functional Profile

The profile of the mission against the Attributes, Secondary Attributes and Factors and the individual systems and SoS configurations are shown in Appendix I.3. The research to date had established an initial set of Attributes, Secondary Attributes and Factors, as shown in Appendix G. No methodology has been established at this stage of how to combine the various measures of the Factors into Secondary Attributes and again how to combine these into Attributes. For this early implementation a simple approach was taken whereby the hierarchy would be shown but the measures would only be applied at the Factor level. The decision maker would then be able to see any mismatches at the Factor level and to which Secondary Attributes/Attributes they contributed. The issue of how to combine factors up into secondary attributes and attributes would be a focus of the next stage of the research. Again the assessment of the mission and systems against the factors was conducted by the researcher as no SMEs were available to support this activity. Whilst the accuracy of the measures can therefore be questioned it is important to emphasise that the value of this research was showing that the approach developed was usable and informative, albeit subject to the quality of the input data.

4.3.1.4 Case Study One Physical Environment Profile

The profile of the mission against the physical environment and the individual systems and SoS configurations are shown in Appendix I.4. The mission was profiled against this established set of metrics to ascertain whether the mission would require the systems to operate in those conditions. The systems were profiled and it was found that they could operate in the physical environment demanded by the mission. This should be the case because as military systems they have been built and trained to operate in a diverse range of climates and environments.

4.3.1.5 Case Study One SoS Profile

When the results of the previous set of characteristics were combined the outcome showed that only the SoS with all four available systems could successfully complete the mission, which concurred with the historical narrative.

4.3.2 Case Study Two – Rescue Of Lt Devon Jones

This subsection is support by Appendix J, Case Study Two – Rescue Of Lt Devon Jones. This case study considers a historical CSAR rescue of the rescue of Lt Devon Jones, an F-14 pilot downed near Baghdad, Iraq, during the 1991 Gulf War. A brief synopsis of the rescue follows.

4.3.2.1 Case Study Two Narrative

This synopsis, culled from (Evans 1999, 133-136, National Museum of the USAF n.d., Pokrant 1999, 31-32), was provided as an example of a CSAR mission in subsection 2.5.2 and is repeated here for the readers benefit. This was the first rescue of a downed airman during Operation Desert Storm.

Lt Devon Jones and his Radar Intercept Officer, Lt Larry Slade, had been tasked with flying their F-14A+ Tomcat, call-sign "Slate 46", on an armed escort mission to protect an EA-6B Prowler aircraft around the airfield at Al Asad, Iraq. At 6.05am whilst returning to their base, approximately 30 miles from Baghdad, Slate 46 was hit by a SA-2 SAM (Surface to Air Missile) and both Jones and Slade ejected at approximately 10,000 feet. Slade used his AN/PRC-112 survival radio whilst parachuting down to call in a mayday message to the orbiting AWACS airplane. The AN/PRC-112 survival radio provides Army Search and Rescue (SAR) personnel with the capability to perform combat search and rescue (CSAR) missions of downed aircrew personnel. As Jones and Slade descended by parachute they were separated in the clouds and darkness and landed separately, some distance apart.

Lt. Jones dug in and camouflaged himself. Meanwhile a MH-53J 'Pave Low' helicopter scrambled to crash site. Overhead Iraqi MiG-23 fighters appeared and were engaged by two USAF F-15Cs on RESCAP (Rescue Combat Air Patrol), the MiG-23's retreat. At approximately 10:30 Lt. Jones hid from what he thought were enemy fighters, but which were actually the two F-15Cs on RESCAP. At approximately 10:30 Lt. Slade was discovered and detained by Iraqi forces. The MH-53J 'Pave Low' helicopter returned to Arar to refuel, returning immediately to continue the search. The 'Sandy' A-10A made contact with Lt. Jones via radio on SAR frequency. Lt. Jones guided the 'Sandy' A-10A to him and the 'Sandy' A-10A located him exactly using its Inertial Navigation System (INS). Iraqi MiG-23s moved into the area and were engaged by the two USAF F-15Cs on RESCAP, the MiG-23's retreated. The 'Sandy' A-10 sanitised the area around Lt. Jones. The MH-53J 'Pave Low' moved in to rescue Lt. Jones, and spotted an Iraqi truck en route to Lt. Jones' position. The 'Sandy' A-10A engaged the Iraqi truck. MH-53J 'Pave Low' landed near the destroyed Iraqi truck, Lt. Jones broke cover and boarded the helicopter, which returned to base. 'Sandy' A-10A's stayed on station to assure the MH-53J's safety. The step by step events of the historical case study are presented in Appendix J.1, Sequence of Events.

4.3.2.2 Case Study Two Functional Profile

The functional profiles are presented in Appendix J.2. The functional profiles have been taken from the developed GFM. The four systems profiled for this case study are:

- A MH-53J 'Pave Low' helicopter (functionally profiled in Appendix J.2.1)
- An F-15C 'Eagle' tactical fighter aircraft (functionally profiled in Appendix J.2.2)
- A 'Sandy' A10A close support aircraft (functionally profiled in Appendix J.2.3)
- The USAF Rescue Coordination Centre (functionally profiled in Appendix J.2.4)

These systems were profiled by the researcher based on available information. As with the previous case study a subject matter expert was sought to aid with this process but none were available through the SEAS DTC. The profiled systems were combined into all possible SoS configurations, as shown in the functional matrix against the missions functional profile in Appendix J.2.5.

When combined into the various SoS configurations with a simple pass/fail indication of the combined as discussed the results showed that both the SoS configuration with all of the available systems and the SoS configuration with the Pave Low, Sandy A10A and USAF Rescue Coordination Centre were functionally capable of conducting the mission.

4.3.2.3 Case Study Two Non-Functional Profile

The profile of the mission against the Attributes, Secondary Attributes and Factors and the individual systems and SoS configurations are shown in Appendix J.3. As described previously the measures were made only at the Factors level for the assessment of the systems which were then automatically combined into the various SoS combination configurations.

4.3.2.4 Case Study Two Physical Environment Profile

The profile of the mission against the physical environment and the individual systems and SoS configurations are shown in Appendix J.4. The mission was profiled against this established set of metrics to ascertain whether the mission would require the systems to operate in those conditions. The systems were profiled and, as with the previous case study it was found that they could operate in the physical environment demanded by the mission. Again, it was realised this should be the case because as military systems they have been built and trained to operate in a diverse range of climates and environments.

4.3.2.5 Case Study Two SoS Profile

When the results of the previous measures were combined the outcome showed that either the combination of all four available systems or the combination of the Pave Low, Sandy A10A and USAF Rescue Coordination Centre could successfully complete the mission. This result is arguably incorrect as considering the historical narrative to the mission the functionality of the F15C's was a decisive factor in the mission's success.

4.4 Evaluation of Prototype Software Tool

A first prototype DSS software tool has been developed and tested using a set of historical CSAR case studies. This section evaluates the case studies in terms of the objectives of the exercise; namely the demonstration of:

- profiling of missions and systems using common sets of characteristics
- formation of SoS profiles from the system profiles
- matching of mission and SoS profiles
- utility and value of the output of this exercise to a military decision maker.

The following subsections consider these in turn before concluding with a way forward for this research.

4.4.1 Profiling Missions and Systems

This section captures the main points identified during the course of conducting the case studies regarding the profiling of missions and systems.

An issue was identified with how the profiles are generated. The case studies found that the system of systems used for each mission were functionally and non-functionally capable of the mission's requirements. The first case study could not have been achieved with a system of systems comprising of a subset of the systems originally used, whereas the second case study could have been completed due to a duplication of required capabilities in the systems. However, this is not a clear conclusion as the findings of the second set of case studies was inconsistent with the historical narrative of the mission. The indication from this is that the profiles used do not capture the key elements of the mission. Of particular concern is the functional set of characteristics. One of the difficulties was profiling the functionality to each system. The functional allocation of SoS driving the developed approach is unable to consider precedence, interoperation and concurrency between component systems. This inability means that the SoS cannot be considered from an open viewpoint which the research problem requires. The VPA methodology used to profile the functionality required by the missions has intertwined functionality with implementation. A more robust methodology that disentangles the functionality from implementation needs to be found.

The second finding was that the Physical Environment characteristics were of little benefit in either case study. Still, as a decision support system these characteristics are significant in that if the decision maker should get it wrong it could have serious implications for the conduct of the mission. These should still be considered as a guard function against poor system selections.

4.4.2 Forming SoS Profiles from System Profiles

The current methodology is exceedingly simplistic in combining systems into SoS configurations. An improved SoS assessment methodology must be found to overcome the current limitation that for a system to fail against a metric, all of the constituent systems must fail. In addition the relationships between systems must be better identified to allow more precise assembly of SoS and identification of key relationships required to conduct CSAR missions. This requires a significant shift in the methodology employed thus far in this research.

4.4.3 Matching of Mission and SoS Profiles

The matching of the mission profile to available SoS profiles seemed clumsy and long winded. There is a need to differentiate between *in theatre* parts of the system of systems and *remote parts*. This is important as certain metrics (especially in the Non-Functional and Physical Environment classes) will only apply to *in theatre* parts of the system. The current assessment method only requires part of the system of systems to have the required capability (and hence not necessarily the part in theatre).

4.4.4 Utility and Value of the Developed Approach

The decision that the developed DSS supports is complex and it needs to present the decision maker with an accurate and clear picture to help inform rather than dictate. Current military decision making processes, such as the Military Decision Making Process (MDMP) do not provide clear visibility of the capability of system of systems to the decision maker, as previously discussed in section 2.3.2. Whilst simplistic the developed DSS does provide clear indications of the suitability of each system of systems, in capability terms, against mission requirements. The transition from theoretical to the practical implementation of the DSS was deemed to be a partial success by the researcher insofar that the tool provided some useful insights and visibility of the system of systems performance against mission requirements. However, the DSS also provided a number of examples of questionable or factually incorrect information. This was compounded as the generated profiles are verbose and do not allow assessment 'at a glance' as RPD would require.

An improved graphical representation of how each SoS profile compares to the mission profile is required, perhaps as a footprint/radar plot as shown in Figure 3 on page 15.

4.4.5 Reconsidering an Approach for a Decision Support System

Some problems have been identified with the approach for a DSS developed so far in this research. The approach relies on three sets of common characteristics: functional, non-functional and physical environment. For the approach to be implemented profiles of individual systems must be combined into SoS profiles due to the time required to assess each SoS separately which would not be fast enough to support the RPD model of decision making. The developed approach utilises disjunction to evaluate across systems to form SoS profiles for each of the sets of characteristics and conjunction to assess across systems as a whole. However, this is an inherently closed viewpoint and fails to capture or address the relationships between systems.

While value has been found in the develop sets of common characteristics a different approach is required in order to address the research problem. An approach based on models as opposed to a parametric approach has been identified as a potential way forward to support the required open viewpoint. A model driven approach may be a more rigorous and repeatable way of addressing the required open viewpoint. The functional profiling is of particular concern and will be the focus of the next architecture. The Non Functional and Physical Environment characteristics will not be considered, but as the components developed to date are reusable they can be integrated into the updated approach at a later stage. The next chapter considers what such a model driven approach might be in the context of this research.

4.5 Chapter Summary

This Chapter has documented the implementation of an approach for a decision support system. To achieve this the architecture had to be developed to allow it to be implemented. This required a shift from the SoS to the system viewpoint, so rather than assessing whole SoS configurations only the individual component systems would be assessed and then assembled into all possible SoS configurations. The implementation of the software tool in Microsoft® Excel® was detailed, including the mechanism for profiling missions and systems within the tool. With the profiling established the method of assessment of SoS profiles against mission profiles was discussed. A set of two historical case studies was conducted using the developed tool. These case studies were evaluated and a number of issues were found. The overall concern with the developed parametric approach was the assembly of systems into SoS and the root of the problem was that the functionality had become entangled with the systems. Due to this coupling there was a lack of relationships between component systems and hence these were not taken into account. A more robust methodology that disentangles the functionality from implementation needed to be found. The model driven approach has been identified as a potential way forward to achieve the required open viewpoint and address the SoS relationship problems. The next Chapter considers such a model driven approach.

Chapter 5

A Model Driven Approach for a Decision Support System

Outline of Chapter

This chapter presents the concepts that will be used to address the issues found in the previous chapter with the original parametric approach. This chapter develops a model driven approach to address these issues. The chapter starts by discussing the structure of language and introduces logical modelling. Model Driven Architecture is discussed, particularly the hierarchical model structure and the role of model transformations. A mathematical approach to model transformations is discussed and incorporated into a integrated model driven approach for this research.

Research Contributions of Chapter

The move towards a model driven approach requires a fundamental shift in this research and hence not all of the research contribution logical model shown in the previous chapters for the parametric approach can be reused. The model driven approach is revolutionary as opposed to the evolutionary progress shown in this thesis so far. Hence the model built up over the last few chapters around the parametric approach is reduced back down to a core model around which the model driven approach will be built on over the next chapters.

*This **Research** established a **Problem** and **Objectives** to develop an **Architecture** of a **Model Driven Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics** which informs the decision maker.*

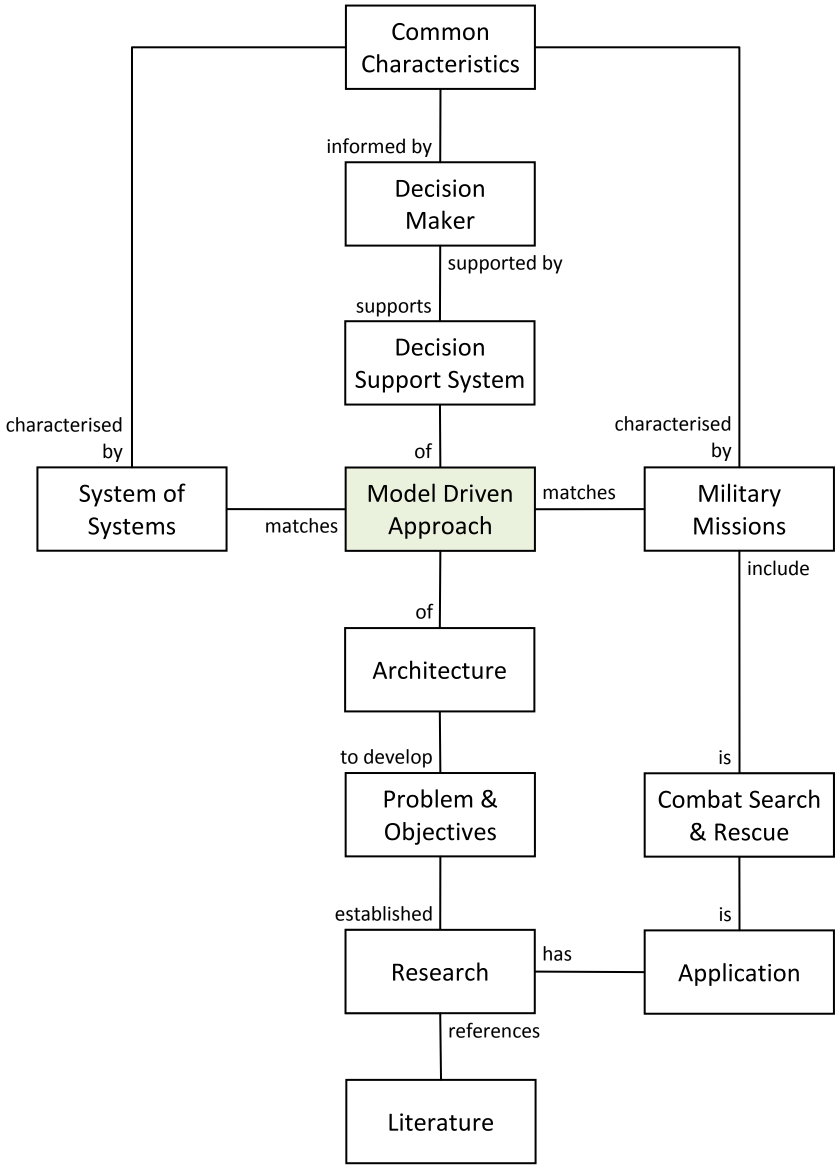


Figure 37 – Logical Model of Research Contributions of Chapter Five

5.1 Developing a Model Driven Approach

The previous chapter established the need for an alternative approach for a decision support system. The approach for a DSS developed to date had addressed the functional, non functional and physical environment characteristics of the missions and systems/SoS. The most pressing issues to resolve was the separation of functional from implementation (which the original VPA methodology had intertwined) and the consideration of the SoS from an open viewpoint, particularly precedence, interoperation and concurrency between component systems. A model driven approach has been identified as potentially overcoming the issues found with the parametric approach developed so far in this research. This chapter starts by introducing the key concepts which will be combined into the model driven approach that the second prototype tool will implement. A model driven approach requires precision (provided by the model) and comprehensibility, which is normally expressed in natural language. This chapter begins by considering the structure of natural language.

5.2 The Structure of Language

This subsection introduces the key concepts of transformational grammar which provides a foundation for subsequent sections. Humans communicate audibly and in writing using natural language, such as English. Linguists have found that our "ability and experience in using our language system to represent and communicate is so extensive that we are able to reflect on the process itself to the extent that we have consistent intuitions about that process" (Bandler and Grinder 1975, 25). For our discussion to follow there are three categories of linguistic intuitions to be aware of:

[redacted]

(Bandler and Grinder 1975, 25-27)

These consistent intuitions have been captured in a meta-model, which is the model of transformational grammar. Transformational grammar is a specific approach to give a set of rules that will correctly predict which combinations of words will form grammatical sentences in natural language. Such approaches are known as generative grammar in theoretical linguistics (Chomsky 1966, 51). Transformational grammar has its roots in Chomsky's work (Chomsky 1957) where he presented his idea that natural language sentences have "linguistic levels" (Chomsky 1957, 11) with two levels of representation – the surface structure which is how natural language is communicated through speech or writing, and the deep structure which represents the core semantic relations of a sentence. The meta-model allows the deep structure to be elicited from a natural language sentence by following the three linguistic intuitions previously presented. Consider the sentence "the man bought a car." We can identify this sentence as fitting the first linguistic intuition of well-formedness. The second linguistic intuition, the constituent structure of the sentence can be revealed as "the man / bought / a car". This is shown in linguistics as a tree structure and is the surface structure of the sentence, as shown in Figure 38.

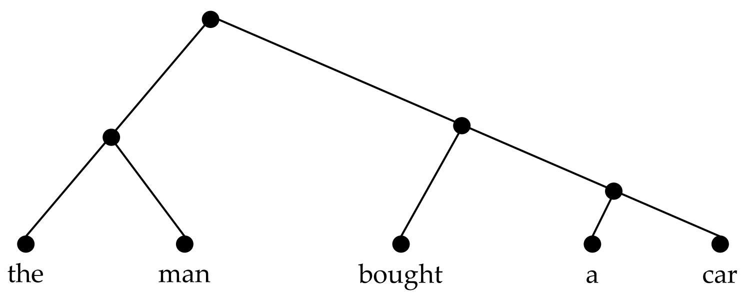


Figure 38 – Sentence Surface Structure

With the surface structure captured as a tree structure the deep structure can be elicited by following our third linguistic intuition as to what the complete representation of the sentence's meaning or logical semantic relation would be. This is the deep structure of the sentence and is shown as a tree structure in Figure 39.

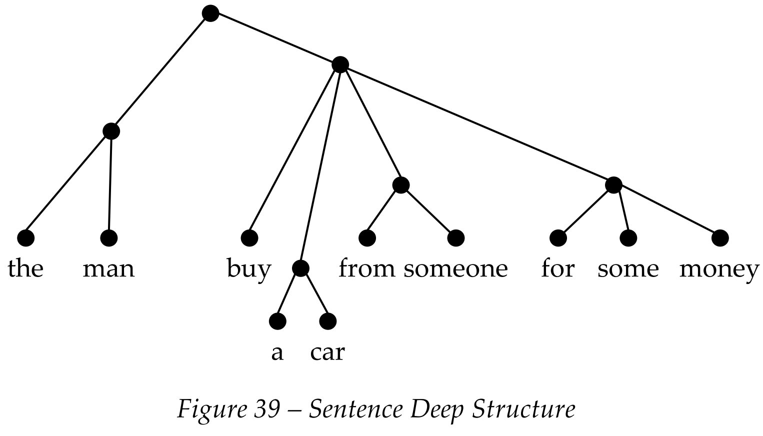


Figure 39 – Sentence Deep Structure

This approach has produced two distinct models of the same sentence – a surface structure model and a deep structure model. Linguists can explicitly state how these two different models are connected through a process called transformation. The transformation states the relationship between the deep structure model and the surface structure model. Transformations between models are a key feature of Model Driven Architecture as will be discussed later. In summary, transformation grammar is "an explicit model of the process of representing and of communicating that representation of the world" (Bandler and Grinder 1975, 37).

How can transformational grammar be applied to an approach for a decision support system? Recognising the intrinsic ambiguity of natural language and the need to move beyond the surface structure considered for the first part of this research, transformational grammar provides a starting point to move towards the precise deep structure. This deep structure is a model and hence transformational grammar provides an approach to create the required deep structure models from a natural language sentence's surface structure. Transformational grammar therefore enables a model driven approach to be utilised by providing an approach to move from the natural language of the original sources for this research to models. Subsection 5.4 will introduce Model Driven Architecture (MDA) which is a model driven approach and it is briefly introduced here to illustrate the link to the transformational grammar presented in this subsection. MDA can be reduced to a simple pattern as Richard Mark Soley, Chairman and CEO of the Object Management Group (OMG), shows below:

[redacted]

(Soley 2004, slide 29)

Note that just like transformational grammar, MDA uses "multiple syntaxes" (surface structures), a model which underlies that semantic (deep structure) and transformations to move between these models. If transformational grammar is to be utilised as the input to MDA how should the deep structure models of the sentences be represented? We have seen already how linguists use tree structures, but these are unsuitable for engineering applications. To help address this gap the next subsection introduces Logical Modelling which uses the Unified Modelling Language (UML) to model the sentences and, it shall be seen, provide an input into MDA.

5.3 Logical Modelling

Logical modelling, like the deep structure of transformational grammar, seeks to "extract the relations that comprise the defined term by using a modelling language to derive a minimal model of the relations (i.e. one that adds no new meaning) but that is complete and captures the intended meaning of the term (i.e. all intended relations have been captured)" (C. Dickerson 2008). The modelling language of choice is the Unified Modelling Language (UML) which is used to capture key words of the sentence (nouns) and the relationships between these keywords (verbs). For systems architecting the need to move from the tree structure preferred by linguists to logical models using UML is founded in Brooks' work:

[redacted]

(Brooks 1995, 234 (6.3))

Brooks' quote above is from a software engineering perspective. To move this to a systems architecting perspective we can substitute the word design for concept and the term prose definition for natural language definition. Now just like transformational grammar we have two different models – the formal definition of a concept (the deep structure) and the natural language definition (the surface structure). If the formal definition can be derived from the comprehensible natural language through transformation then the natural language can be used for reasoning. Hence, "conceptual integrity of the formal definition of a specific term is preserved by concordance between precision and comprehensibility" (C. Dickerson 2008). The Logical Modelling approach will be described in detail when it is applied in section 6.3. For now it is suffice to say that it builds on the concept of Transformational Grammar and provides an approach that meets the specific needs of engineering. With an approach identified that can move from source materials into precise models we now consider a model driven approach that can utilise these models.

5.4 Model Driven Architecture

The Model Driven Architecture (MDA) is a standard developed by the OMG (Object Management Group) which has been an international, open membership, not-for-profit computer industry consortium since 1989. The MDA standard is captured in the MDA Guide v1.0.1 (Miller and Mukerji 2003). The OMG viewpoint of the MDA is for software development and is described by them as follows:

[redacted]

(Miller and Mukerji 2003, 2-2)

The above quote captures a key principle of MDA: *separating the specification of the operation of a system from the details of the way that system uses the capabilities of its platform*. Note that this principle is independent of software development. This is an example of how a principle from MDA can be applied to the systems engineering domain that this research addresses. This principle also directly addresses the main issue found with the parametric approach: the entangling of functionality with implementation. The MDA is an approach to systems development that seeks to separate the technology based implementation from the core solution of the problem. This is the separation of function from implementation that the first approach lacked. The aim of this is to establish a platform independent model which captures the functionality required in such a way that the ultimate implementation of that functionality can be migrated across technology. In MDA the term system architecture is defined as follows:

[redacted]

(Miller and Mukerji 2003, 2-3)

MDA has three key viewpoints: the Computational Independent Viewpoint, Platform Independent Viewpoint, and the Platform Specific Viewpoint. The Computational Independent Viewpoint focuses on the environment of the system and requirements for the system. The details of the structure and processing of the system are hidden or as yet undetermined. The Platform Independent Viewpoint focuses on the operation of a system while hiding the details necessary for a particular platform. It shows that part of the complete specification that does not change from one platform to another. The Platform Specific Viewpoint combines the platform independent viewpoint with an additional focus on the detail of the use of a specific platform.

From these viewpoints MDA specifies three models: the Computational Independent Model (CIM), Platform Independent Model (PIM), and the Platform Specific Model (PSM). In UML requirements are captured with use cases which call out high level interactions between the system and its environment. Use cases can be used to identify high level functions that the system must perform. In the model driven approach described in section 5.5 and applied in Chapter 6 section 6.5 the CIM will be captured using high level functions. This is also in line with the GFM developed in the initial approach. The PIM will capture the relationship of the flows between the functions identified in the CIM. This will identify the operation of the SoS while hiding the details necessary for a particular platform.

In MDA the system is specified in the Platform Model (PM) which captures the technical concepts of the system's implementation and represents the different parts of the system and the services it provides. The relationship between these models is illustrated in Figure 40.

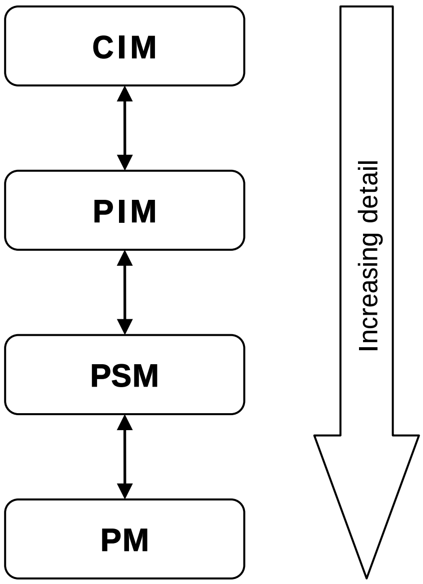


Figure 40 – MDA Model Hierarchy

Whilst these models from the various viewpoints are useful by themselves the real power of MDA is the ability to move between models whilst maintaining the conceptual integrity of the system, this is achieved through model transformations which is focus of the next subsection. The arrows shown in Figure 40 depict the ability to transform between models, which is the subject of the next subsection.

5.4.1 Model Transformations

Model transformation is the key to MDA and "is the process of converting one model to another model of the same system" (Miller and Mukerji 2003, 2-1). This process in MDA is illustrated in Figure 41.

Figure 41 – MDA Model Transformation, adapted from (Miller and Mukerji 2003, 2-7)

Model transformation is enabled through the use of an MDA mapping which "provides specifications for the transformation of a PIM into a PSM for a particular platform" (Miller and Mukerji 2003, 3-2). There are two distinct approaches to mapping: (i) type mappings and (ii) instance mappings. Most mappings normally consist of a combination of these approaches. Type mappings are rule based mappings that are used to specify "a mapping from any model built using types specified in the PIM language to models expressed using types from a PSM language" (Miller and Mukerji 2003, 3-2). As general rule based operators, type mappings are very difficult to define. Hence, instance mappings are more commonly used as they are easier to define. Instance mappings are used to specify how a particular element from the PIM should be transformed into the PSM by using defining marks. A mark "represents a concept in the PSM, and is applied to an element of the PIM, to indicate how that element is to be transformed" (Miller and Mukerji 2003, 3-3). While a type mapping is a general rule based transformation, instance mappings are point to point transformations and hence are easier to define because their scope is very specific. Model transformation can be thought of as part of the design and development process (Raistrick, et al. 2004). The role of transformations in MDA allowing the move from the PIM to PSM and then code (which is the PM) is shown in Figure 42.

Figure 42 – A Simple View of the MDA Approach for Software Development (Jones, et al. 2007)

While MDA prescribed transforms to move between models through mapping, and particular types of mapping, it does not prescribe a mechanism to achieve this in practise. This concept of transformation was introduced with transformational grammar back in subsection 5.2. Just as we sought a more rigorous approach to modelling the surface and deep structures of natural language beyond transformational grammar, now we seek a more rigorous approach to model transformation, which is the focus of the next subsection.

5.4.2 A Mathematical Approach to Model Transformation

This subsection presents a novel way of automating MDA transformations using an approach described in Dickerson and Mavris (2009). The importance of this mathematical approach is that it adds a logical foundation to model transformation, and it will be shown in so doing to enable the maintenance of conceptual integrity throughout the design process. A more detailed explanation of the transformations is presented in Appendix L

The logical modelling approach introduced previously in subsection 5.3 produces a UML model of natural language sentences as its output. This UML diagram explicitly exposes the relationships between the keywords in the natural language sentence. These keywords are system parameters. These relationships can be depicted using a square matrix, such as the $\underline{M}$ matrix in Figure 43. The $\underline{M}$ matrix represents the system parameters from the logical model (expressed as $y_1 \dots y_m$) with the ticks indicating a relationship between the parameters. Note that these relationships are one way and the matrix is read row-column, so for example the bottom left relationship in the $\underline{M}$ matrix indicates that y_{m-1} is related to y_2 (but not that y_2 is related to y_{m-1}). This can be expressed using the notation $y_{m-1}My_2$ which means that $\underline{M}$ has associated y_{m-1} with y_2. The matrix $\underline{M}$ shows that y_2 is related to y_{m-1} by the upper right tick (y_2My_{m-1}).

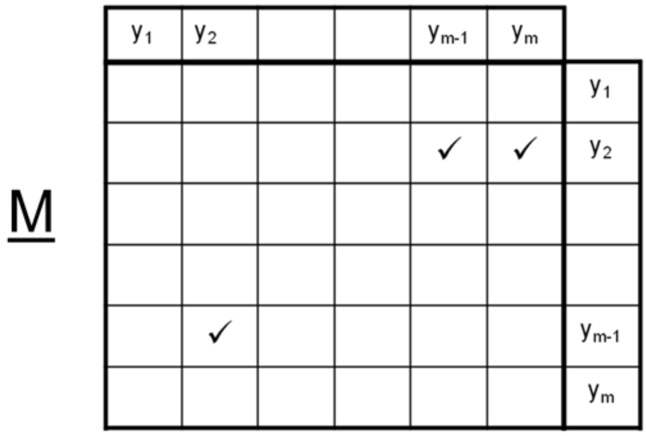


Figure 43 – Relationship Matrix

The MDA transforms one parametric model (with a particular viewpoint of the system) into another parametric model (with a different viewpoint of the system). This transform is specified using instance mappings which are easier to define than their type mapping counterparts, as previously described in subsection 5.4.1. These instance mappings can be specified using another matrix attached to $\underline{M}$ which delineates the mapping of the $y_1 \dots y_m$ onto another set of keywords, for example $x_1 \dots x_n$. This is illustrated in Figure 44 which shows the relationship matrix $\underline{M}$, the mapping matrix Q and a second relationship matrix $\underline{\mathrm{N}}$. The relationships established between $y_1 \dots y_m$ are mapped onto a second set of parametrics ($x_1 \dots x_n$) in Q. By tracing through these relationships the relationships between $x_1 \dots x_n$ can be derived as shown in $\underline{N}$. For example, from the Q matrix we can determine the following relationship:

$$y_{m-1}My_2$$

and mapping y_{m-1} and y_2 onto x using the Q matrix we find:

$$y_{m-1}Qx_3 \text{ and } y_2Qx_2 \text{ or } y_2Qx_n$$

Therefore,

$$\text{If } y_{m-1}My_2 \text{ then } x_3Nx_2 \text{ and } x_3Nx_n$$

Note how two relationships in $\underline{N}$ are produced from one relationship in $\underline{M}$ due to the double relationship of y_2 in Q. Thus with the $\underline{M}$ and Q matrices complete the $\underline{N}$ matrix can be derived from them. Therefore any parametric system model (e.g. $\underline{M}$) can be transformed into another parametric system model (e.g. $\underline{N}$) given a mapping matrix (such as Q). The mathematics behind these matrix operations underline the validity of this approach and is described in more detail in Dickerson and Mavris (2009), a brief summary of which follows. Consider the $\underline{Q}$ matrix again and the notation yQx. This notation can be extended to RQ which describes how Q transforms a subset R of $\underline{M}$ into a subset RQ of $\underline{N}$. The subset R, in general, represents a mathematical relation on the underlying set of parameters. RQ is defined as follows: for each pair of parameters $(y_i,\ y_k)$ that belong to the mathematical relation R, if y_iQx_j and y_kQx_l then the pair $(x_j,\ x_l)$ belongs to the mathematical relation RQ in $\underline{N}$. Figure 44 illustrates such a model transformation. The model $\underline{M}$ has three relationships, i.e. $R = \{(y_2, y_{m-1}), (y_2, y_m), (y_{m-1}, y_2)\}$. Note that the first and third relationships are symmetric.

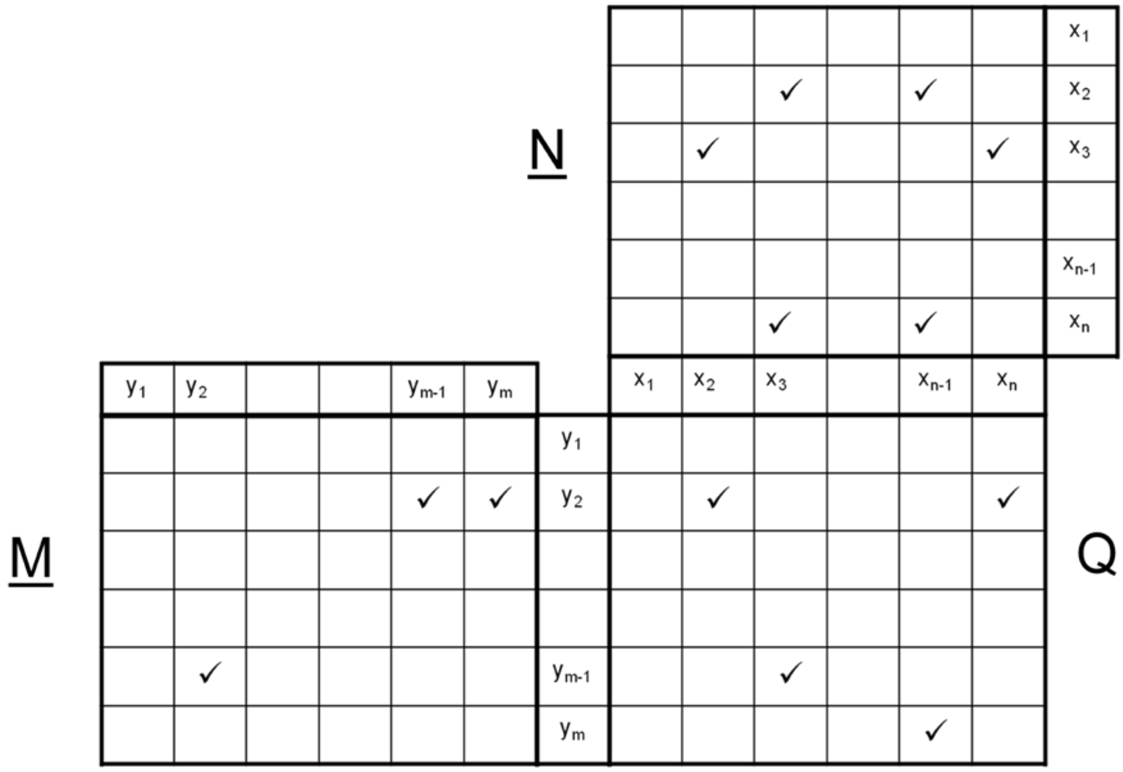


Figure 44 – Relationship and Mapping Matrices

The model transformation Q acts on three values of the parameters in $\underline{M}$: y_2, y_{m-1} and y_m. The action of Q on y_2 is doubled valued, so combined with the symmetric relationship between y_2 and y_{m-1} this maps to two symmetric relationships in the model $\underline{N}$ (a total of four individual relationships). The double-valued action of Q on y_2 and single-valued action on y_m transforms the other relationship that y_2 has with y_m into two relationships in $\underline{N}$. Therefore, Q transforms the three relationships in $\underline{M}$ into the six relationships RQ in $\underline{N}$ depicted in the figure.

Recall that MDA seeks, among other things, to separate the technology of the solution from the processes required by problem. With the model transformation shown in Figure 44 this concept can be given a more precise meaning. The problem (expressed as flows) would be modelled by $\underline{M}$ and the solution (expressed as systems) would be modelled by N. The structure of the problem, as represented by the matrix $\underline{M}$, is *reflected* into the solution $\underline{N}$ by the transformation Q. In Figure 44, at the level of relationships, R is precisely all the pairs (y_i, y_k) that are related in $\underline{M}$, and the *reflection* of R is represented by the mathematical relation RQ in $\underline{N}$.

With the mathematics behind this process and hence its validity established the next logical step to take was to try and automate this transformation process in a software tool. A software tool was developed as a concept demonstrator to show that one system parameter model can be transformed automatically into another system parameter model given a transformation matrix. This software tool was successfully implemented in Microsoft Excel and is described in more detail in the next Chapter 6 in section 6.2. Experimentation with the tool verified that it was transforming the matrices correctly and a screen capture of the software tool is shown in Figure 45.

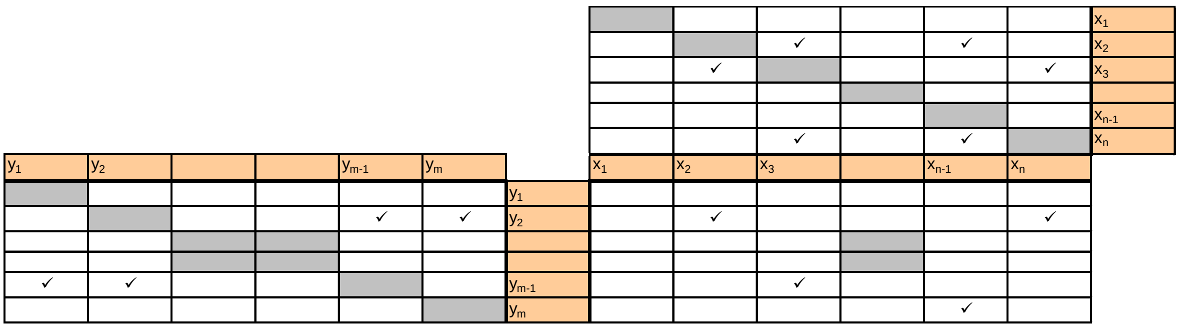


Figure 45 – Matrix Transformational Tool

The successful implementation of the tool provides a way to rapidly perform model transformations. To move this tool beyond automated model transformations and into a more integrated engineering tool it needed its functionality extending to provide deeper insights. An integrated model driven approach utilising the concepts discussed so far was sought, which is discussed in the next section.

5.5 An Integrated Model Driven Approach

This chapter has introduced a number of key concepts: Transformational Grammar, Logical Modelling, Model Driven Architecture and model transformations. In this subsection a coherent and complete approach will be presented pulling together all of the concepts discussed into a complete approach.

Transformational Grammar has been presented as a specific approach of generative grammar in theoretical linguistics. Transformational Grammar identifies two levels of representation of natural language sentences – the surface structure and the deep structure. Linguists can explicitly state how these two models are related through a process called transformation. An approach to elicit the deep structure from the surface structure of natural language in a model usable by an engineering process was required as the tree structure used by linguists was not appropriate. The Logical Modelling approach, which creates both surface structure and deep structure models of natural language using the Unified Modelling Language (UML), was identified as appropriate for this research. In the engineering domain the Model Driven Architecture has been identified as an approach which implements the model transformation concepts of Transformational Grammar for engineering applications. Model Driven Architecture is a realisation of the principles of structured analysis and design and achieves separation of the problem specification from the solution through the logical separation of the Computational Independent Model (CIM)/ Platform Independent Model (PIM) and Platform Specific Model (PSM) models. A lower Platform Model (PM) as specified by the MDA has been considered but is beyond the scope of this research and the approach has been bounded to the CIM/PIM/PSM models. MDA prescribes model transforms to move between these models and we have presented a methodology to achieve this whilst maintaining conceptual integrity through the use of mathematical matrix transforms. This has been captured in a developed software tool. Whilst this process provides a solution in the form of the PSM (or even PM) it does not provide a repeatable input methodology to the CIM. We now present a method for moving from a logical model created from a natural language sentence of the systems requirements into a CIM and in so doing move towards a complete model driven approach.

A logical model is not appropriate for inputting into a CIM as the keywords it has identified are not necessarily system attributes. A methodology is thus required to transform the logical model into a form suitable for inputting into the MDA approach. The methodology developed takes the logical model and transforms it into a flow diagram (Yourdon 1988). The flow diagram contains information for both processes (CIM) and flows (PIM), as illustrated in Figure 46. The flow diagram can then be used to populate the CIM, CIM-PIM Transform and in turn the PIM, as illustrated again in Figure 46. The method will be shown in full when it is applied in the next Chapter 6, section 6.4.

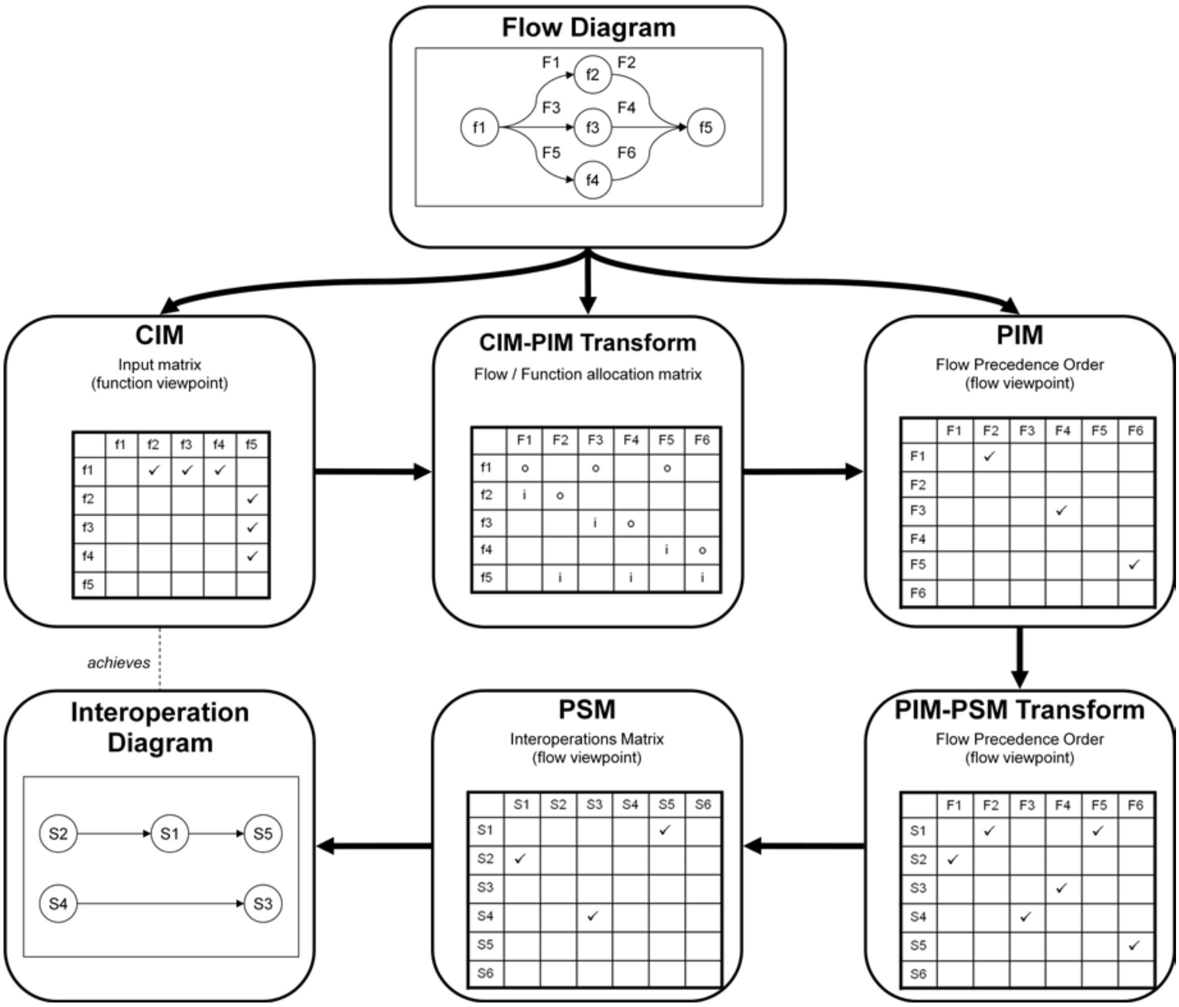


Figure 46 – Flow Diagram Input into MDA

The process starts in Figure 46 with the flow diagram which is determined by the analysis of doctrines and tactics. This will be achieved through the logical modelling of doctrines and tactics in section 6.3 which in turn will be transformed into a flow diagram in section 6.4. The CIM, CIM-PIM Transform and PIM can be determined from the flow diagram and this will be shown in section 6.5. With the set of flows established these can be mapped to systems in the PIM-PSM Transform. This is new information introduced in the form of the commanders intent. This, and the subsequent stages, will be shown for each of the case studies repeated in Chapter 7. The PIM can then be transformed using the PIM-PSM Transform into the PSM from which the interoperation diagram can be determined. Appendix M shows a user manual which details how the process shown in Figure 46 can be used in practice.

Figure 46 illustrates the relationship between the MDA models – the CIM, PIM and PSM. Consider the discussion back in Chapter 1 section 1.2.4 on capability. To recap, capability moves away from a *means* based perspective to a combined *ways* and *means* perspective. Figure 46 shows such a perspective, the *ways* being shown by the CIM and PIM, the *means* by the PSM. This is illustrated in

Figure 47. Hence this approach may help define and deliver mission defined capability requirements.

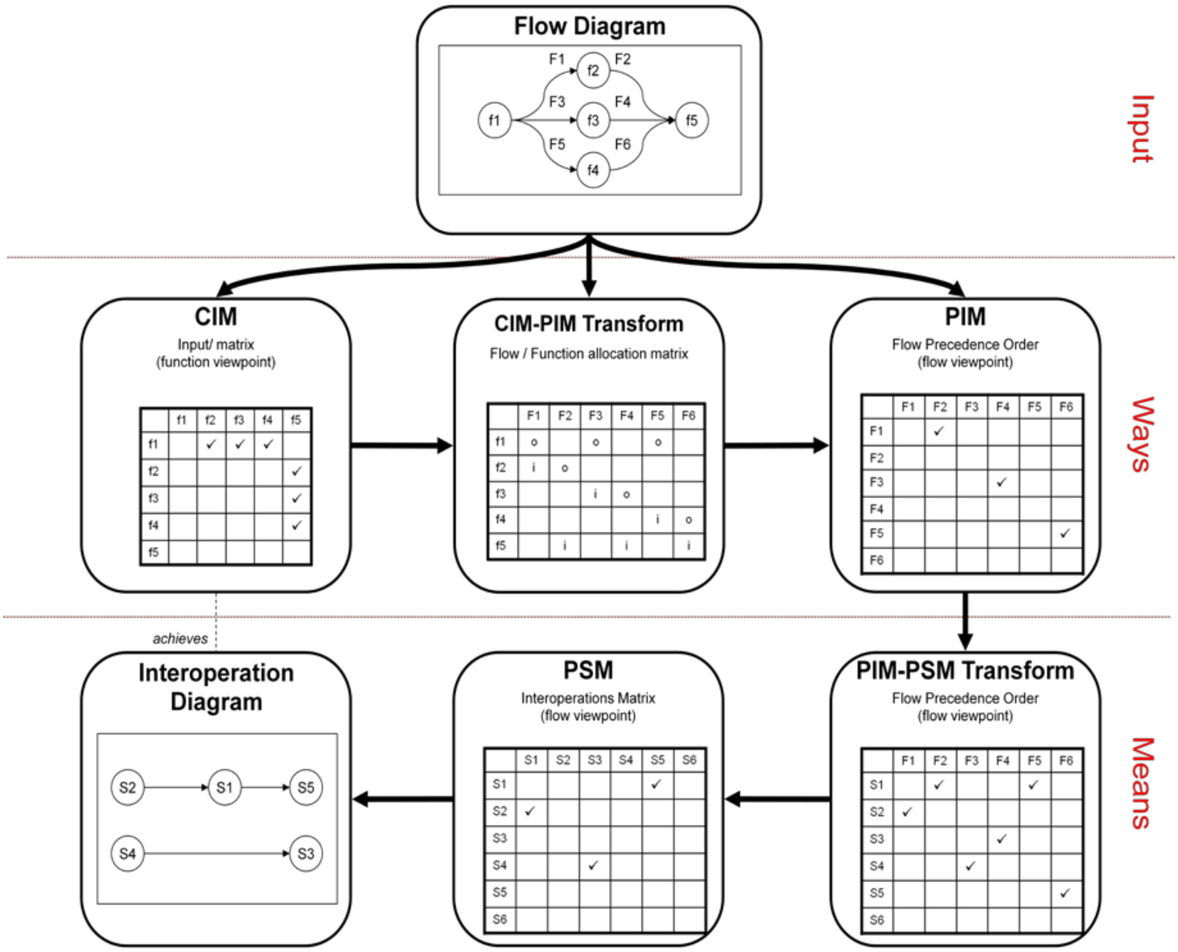


Figure 47 - Flow Diagram Input into MDA showing Input, Ways and Means

With an input from logical models into the MDA approach established the next consideration is how to support the decision maker. The original approach shown previously in Figure 36 and repeated here in Figure 48 for readability was a linear parametric approach.

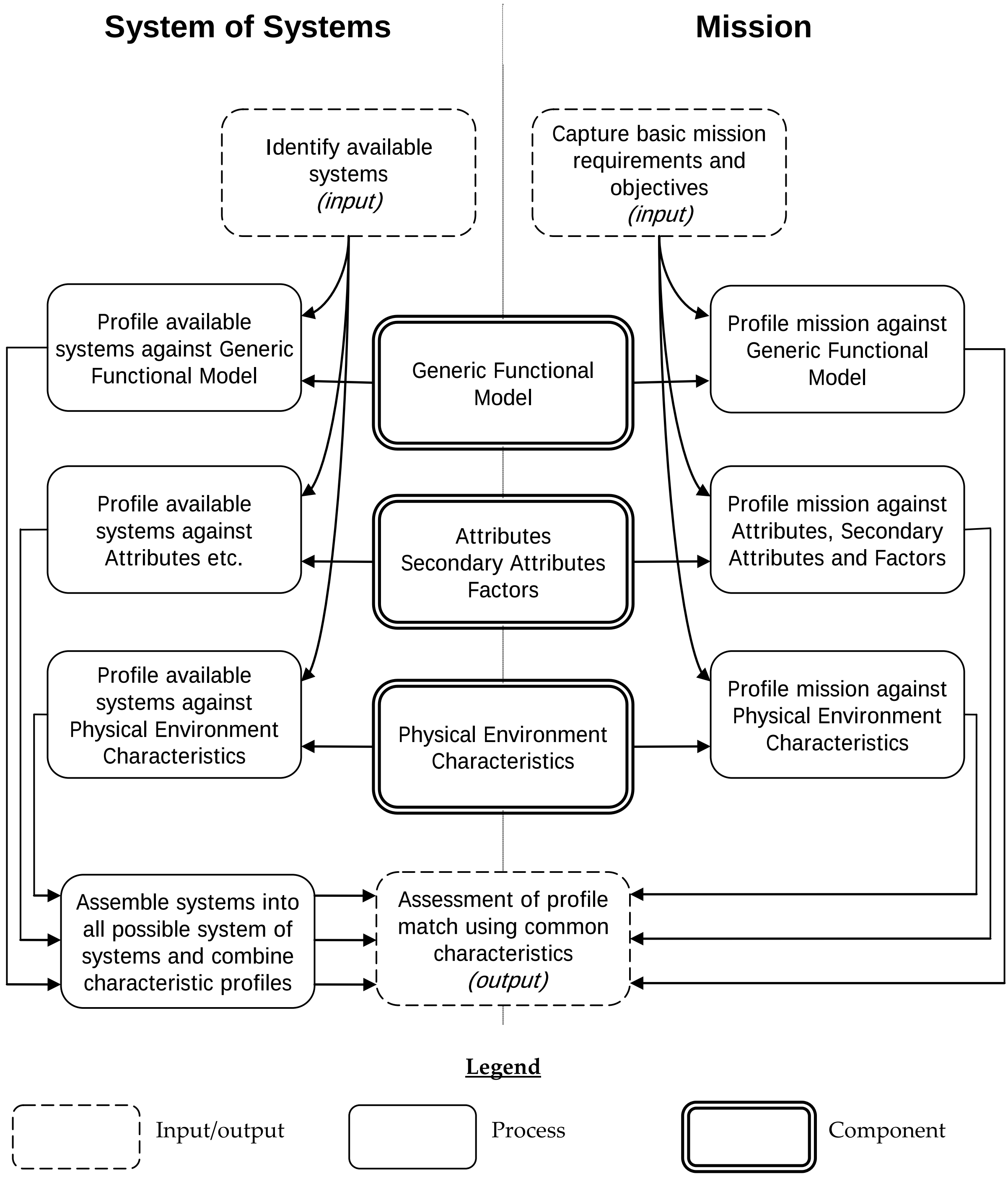


Figure 48 – Implemented Architecture of an Approach for a DSS

The available systems and the mission were profiled against each of the developed common sets of characteristics in turn, the system profiles were combined simplistically into SoS profiles and these were then compared to the mission profile. The lack of consideration of relationships between component systems and the intertwining of functions with implementation were issues with this first approach. The model driven approach combining the elements discussed in this chapter is shown in Figure 49.

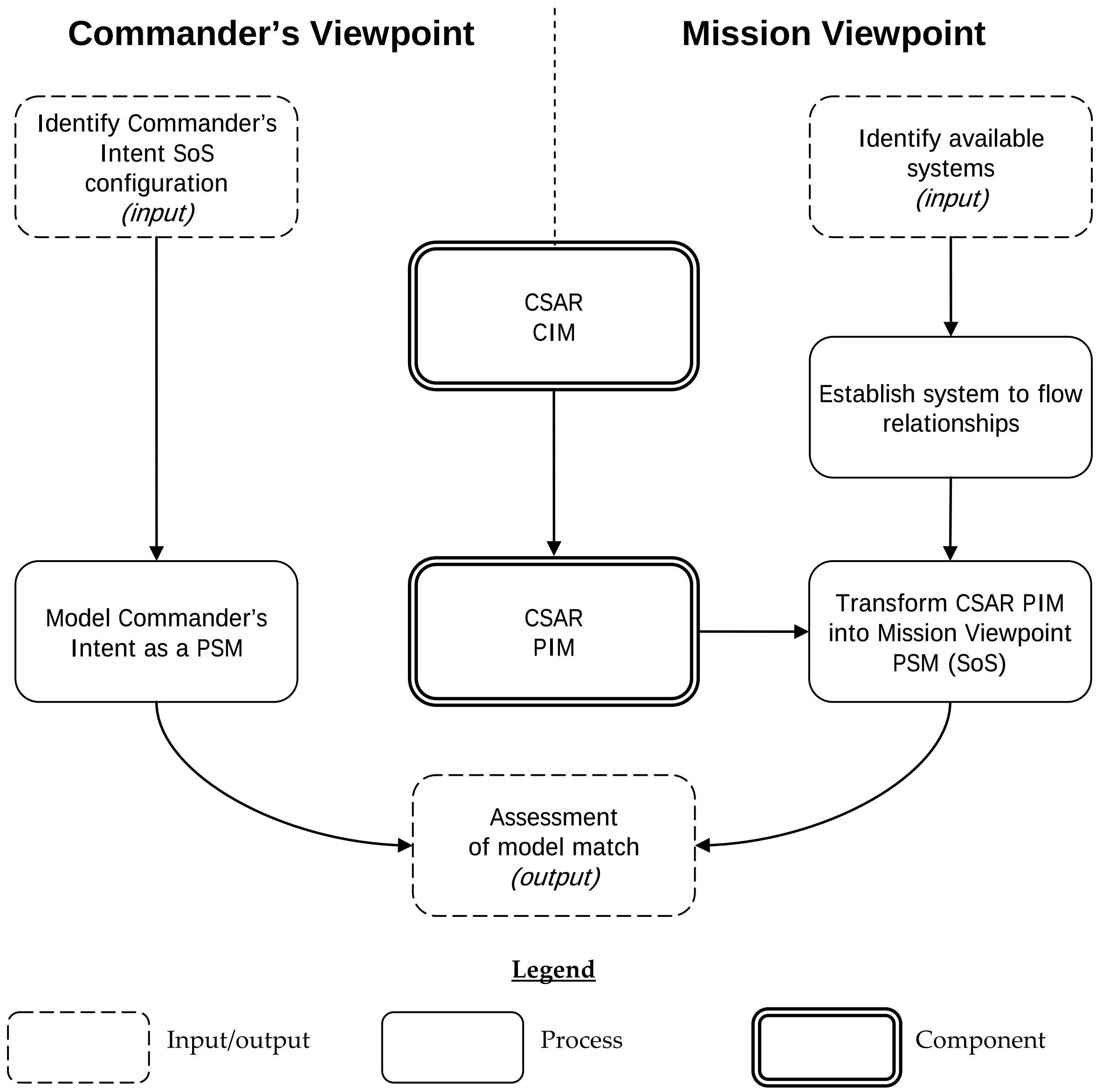


Figure 49 – An Architecture of a Model Driven Approach for a DSS Utilising CIM, PIM, PSM

The model driven approach captures the Commanders Intent and identifies the available systems. The Commanders Intent is modelled as a PSM (SoS). The available systems to flow relationships are established and, using the CSAR PIM, the PIM is transformed into a Mission Viewpoint PSM (SoS). The Commanders Intent PSM and Mission Viewpoint PSM are compared and through this comparison an assessment of the model match is made.

The main advantages of the model driven approach is that it is a rigorous and repeatable and removes much of the inherent ambiguity contained in the previous approach. It allows an open viewpoint to be taken through the capturing of relationships and the transformation down from the conceptual level rather than assembly up from the implementation level. Open viewpoints are of course wicked and the developed model driven approach helps address this by bounding the higher conceptual level through its representation as processes and flows with defined relationships between them. These relationships and the ability to transform between model levels help ensure the conceptual integrity of the approach. The ability to bound the problem through an independent model representation that can be transformed into interoperable solutions allows an open viewpoint to be maintained.

Note that in the model driven approach only the functional characteristics are considered as opposed to the functional, non-functional and physical environment characteristics considered in the previous parametric approach.

The development of this approach which forms the basis of the second implementation of an approach for a decision support system is presented in the next Chapter 6. Consider again the original statement of the problem that this research addresses:

Is there a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission?

Statement of Research Problem

The original aims of this research were to provide a decision support tool for decision makers to help them decide which SoS to use for a particular CSAR mission. This approach aids that by allowing automated generation of SoS through model transformation of the mission requirements while preserving an open viewpoint. In addition it allows automated comparison of available SoS options against the mission requirement through comparative transformations. Therefore, this software tool builds on the first prototype DSS software tool by providing a methodology that ensures conceptual integrity through an MDA approach whilst minimising the decision makers workload by using automated model transforms for both option generation and comparison. This integrated model driven approach will be further elaborated upon in Chapter 6.

5.6 Chapter Summary

This Chapter has introduced a number of key concepts: Transformational Grammar, Logical Modelling, Model Driven Architecture and model transformations. Transformational Grammar has been presented as a specific approach of generative grammar in theoretical linguistics. Transformational Grammar identifies two levels of representation of natural language sentences – the surface structure and the deep structure. An approach to elicit the deep structure from the surface structure of natural language in a model usable by an engineering process was required as the tree structure used by linguists was not appropriate. The Logical Modelling approach, which creates both surface structure and deep structure models of natural language using the Unified Modelling Language (UML), was identified as appropriate for this research. In the engineering domain the Model Driven Architecture has been identified as an approach which implements the model transformation concepts of Transformational Grammar for engineering applications. Model Driven Architecture is a realisation of the principles of structured analysis and design and achieves separation of the problem specification from the solution through the logical separation of the Computational Independent Model (CIM)/ Platform Independent Model (PIM) and Platform Specific Model (PSM) models. This separation of the operation of a system from the details of the way that system uses the capabilities of its platform addresses the issue found with the parametric approach where the functionality and implementation had become intertwined.

A lower Platform Model (PM) as specified by the MDA has been considered but is beyond the scope of this research and the approach has been bounded to the CIM/PIM/PSM models. MDA prescribes model transforms to move between these models and we have presented a methodology to achieve this whilst maintaining conceptual integrity through the use of mathematical matrix transforms. This has been captured in a developed software tool. Whilst this process provides a solution in the form of the PSM (or even PM) it does not provide a repeatable input methodology to the CIM. A method to achieve this has been presented to allow Logical Models to be converted into flow diagrams which in turn can be converted into a CIM and PIM. A coherent and complete approach pulling together all of the concepts discussed has been presented which forms a model driven approach. The next Chapter 6 discusses the implementation of the architecture of this approach.

Chapter 6

Second Implementation of an Approach for a Decision Support System

Outline of Chapter

This chapter presents the development and implementation of the second prototype Decision Support System (DSS) software tool. This software tool implements the model driven approach introduced in the previous Chapter 5 which seeks to disentangle functionality from implementation, an issue with the first approach for a DSS. To enable the tool's development a logical model of combat search and missions is created from available doctrines. This logical model is converted to a flow diagram from which a CIM and PIM are identified for combat search and rescue missions. The methodology for profiling systems is presented and finally the evaluation process is discussed.

Research Contributions of Chapter

From the previous chapter(s):

*This **Research** established a **Problem** and **Objectives** to develop an **Architecture** of a **Model Driven Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics** which informs the decision maker.*

This chapter introduces:

*In the model driven approach the system of systems is captured in a **Platform Specific Model**, the common characteristics are captured in a **Platform Independent Model** and the combat search and rescue mission is captured in a **Computational Independent Model**. These models can be related by **Model Transformation**. The architecture has been implemented in a **Software Tool**.*

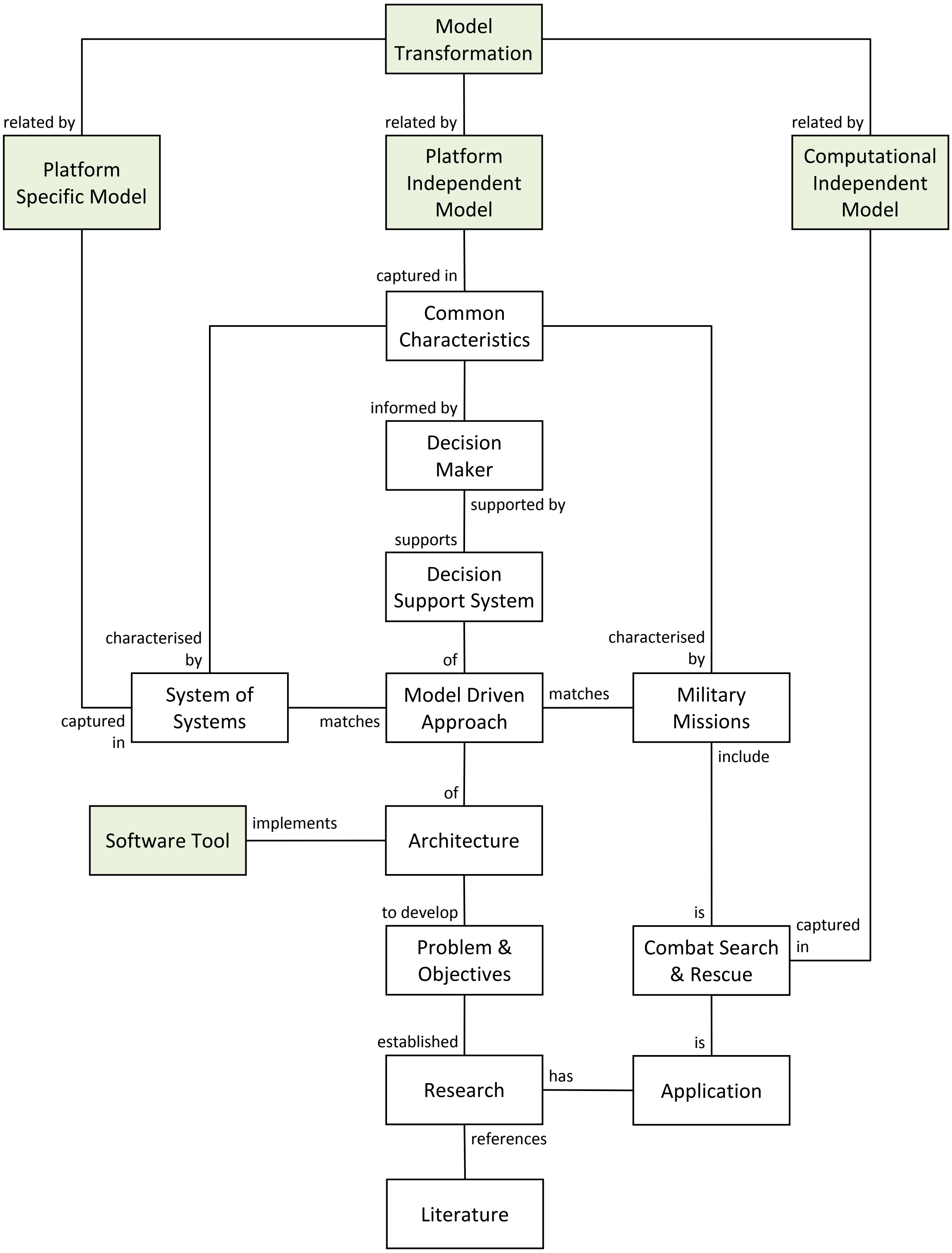


Figure 50 – Logical Model of Research Contributions of Chapter Six

6.1 Second Implementation Introduction

This chapter documents the implementation of the revised architecture of a model driven approach for a decision support system. Like the first tool this second prototype tool will be used to demonstrate that the concepts presented can be implemented in a practical tool which will be used to conduct a second set of case studies. This chapter aims to realise this architecture in a prototype software tool.

Three main aims were set to be achieved by developing this second prototype software tool:

1. **Feasibility** - demonstrate that the second developed DSS architecture could be implemented in a software tool.
2. **Validation** - perform a set of historical case studies utilising the developed second prototype software tool.
3. **Guide** - guide the further development of the architecture of an approach for a DSS.

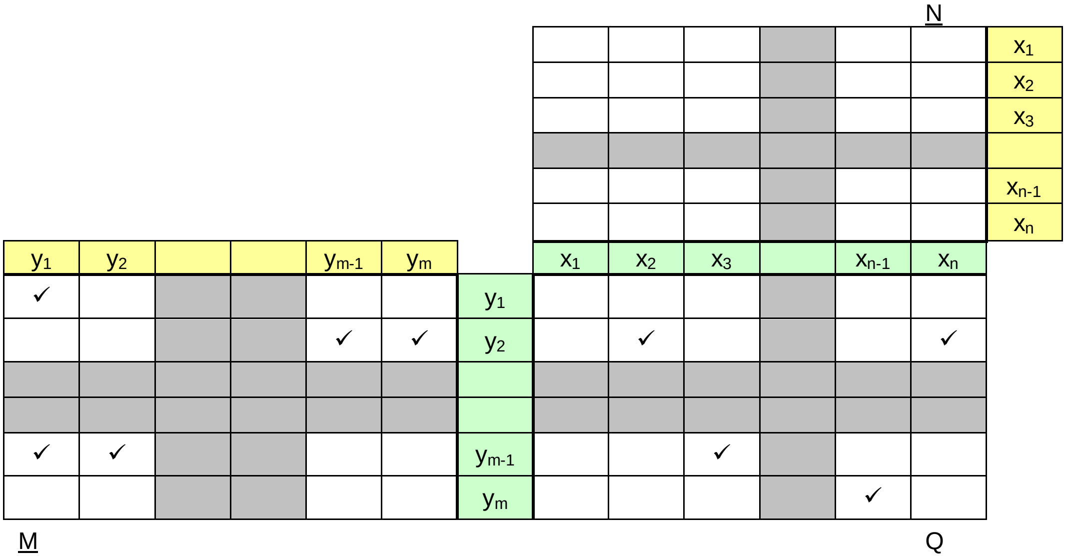
These aims will be addressed through the development of the tool in this Chapter and the evaluation of the tool in the next Chapter 7.

6.2 Developing a Second Prototype Decision Support System Software Tool

The development of the second prototype DSS software tool presented many challenges which had to be met to achieve the basic implementation that this research sought. The initial motivation for the tool was to simply automate model transformation.

The first challenge was to develop a software tool which would automate the process of model transformation from one model matrix and a transform matrix into another model matrix. Take the example illustrated below in Matrix 1 which has a model matrix ($\underline{M}$) and a transform matrix (Q) completed.

Matrix 1 – A Model Transformation Matrix from M to N Transformed by Q

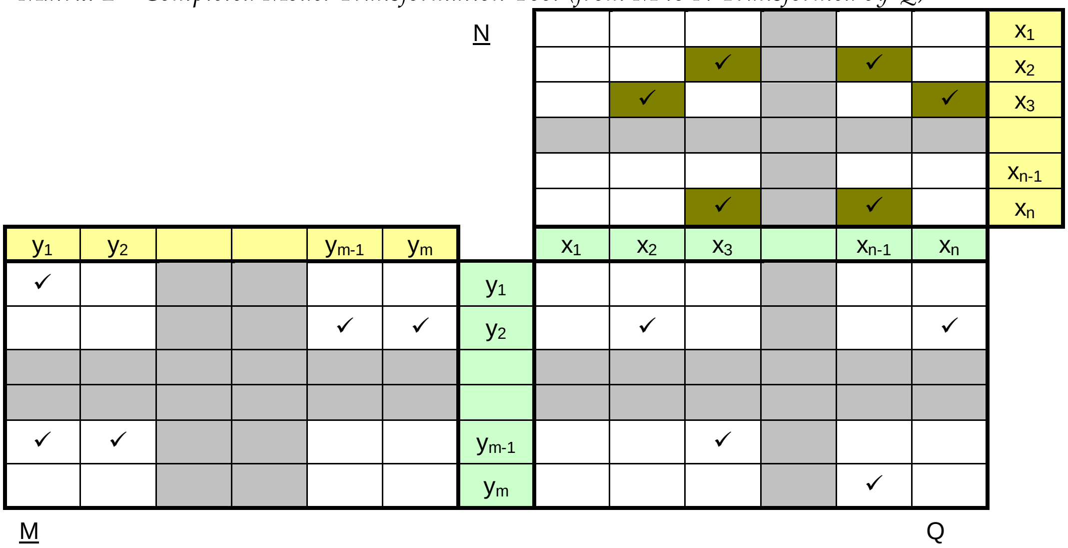


The software tool has been developed in Microsoft Excel 2007 and uses Visual Basic code to automate the transformation. The software follows the following steps to transform $\underline{M}$ to $\underline{N}$ through Q:

- Determine relationships in $\underline{M}$
- Determine relationships in Q
- Scan through $\underline{M}$ relationships and for each match to relationships in Q. If a mapping can be found capture in the $\underline{N}$ relationship.

This software tool was enhanced with a feature that allowed the $\underline{N}$ matrix to either be determined or, if prefilled, to check the accuracy of $\underline{N}$. To enable the accuracy to be checked code was written to shade in the cells in green (correct) or red (incorrect or, if blank, missing relationship). The updated tool and the cell shading on $\underline{N}$ can be seen in Matrix 2.

Matrix 2 – Completed Model Transformation Tool (from M to N Transformed by Q)



This initial developed tool identified an interesting feature of model transformation, that of the degree of the relationship imposed by the transformation. Each relationship in $\underline{N}$ may be unique (created by a single mapping of $\underline{M}$ via Q) or may be shared ($\underline{M}$ and Q create multiple relationships on a single relationship in $\underline{N}$). Unique relationships are not a problem, but shared relationships are as the generated relationships need not necessarily be the same. The number of generated relationships on a shared relationship is expressed as degrees (so two degrees would imply that two generated relationships were put on a single relationship in $\underline{N}$. Such conflicts need to be at least highlighted to allow resolution. To achieve this a warning indicator was implemented in the code to highlight such conflicts. The issue of degrees will be addressed more thoroughly in a later version of the tools development.

An observation of this initial software tool is that while $\underline{N}$ can be generated from $\underline{M}$ and Q there is an opportunity to include an additional transformation matrix to allow $\underline{M}$ to be generated from $\underline{N}$. Q cannot be used to transform $\underline{N}$ to $\underline{M}$ as the relationships captured within Q are not necessarily symmetrical. Hence another transform matrix is required (additionally this also maintains the row/column reading formalism for these matrices). The inclusion of this additional transformation matrix (P) completes the transformation circle from $\underline{M}$ to $\underline{N}$ and back to $\underline{M}$ and is shown in Matrix 3.

Matrix 3 – Extended Model Transformation Tool

P (top left), N (top right), M (bottom left), Q (bottom right)

						x_1						
						x_2			✓		✓	
						x_3		✓				✓
						x_{n-1}						
						x_n			✓		✓	
y_1	y_2			y_{m-1}	y_m		x_1	x_2	x_3		x_{n-1}	x_n
						y_1						
				✓	✓	y_2		✓				✓
✓	✓					y_{m-1}			✓			
						y_m					✓	

This allows either $\underline{N}$ to be derived from $\underline{M}$ and Q or $\underline{M}$ to be derived from $\underline{N}$ and P, but it also allows the resultant model to be reflected back onto the original model to understand any implications or conflicts that could result from the generated relationships. The code that allows model transformation in the software tool is detailed in Appendix N.

With the underlying code mechanics established a model for CSAR missions had to be created. The GFM created for the previous parametric approach was not usable due to the lack of relations between the functions. A logical model was required, and the creation of such a model is detailed in the next section.

6.3 Logical Modelling of CSAR

This section will outline an approach to deriving a logical model for CSAR from a sentence sourced from the literature that provides the starting model from which the surface structure of CSAR may be derived. This in turn will be used to derive the deep structure of the sentence that can be utilised in a Model Driven Architecture (MDA) approach to provide an input/output matrix which will serve as a CIM-PIM Transform and allow a CIM to be derived from the precedence order established within that matrix. Thus, by extending logical modelling with the transformational grammar theory on which it is based, a formulised and repeatable methodology is presented to allow a sentence to be transformed into a CIM and PIM that can be used for exploration of solutions at the PSM and, eventually, PM levels.

The objective of Logical Modelling is presented through a direct quote:

[redacted]

(C. Dickerson 2008)

To model a term the first step is to identify a sentence from appropriate literature that contains a definition of the term of interest. It may be that sentences defining the term of interest, but with different content and structure, will be available from multiple sources (or may occur within one source). In such cases collecting these sentences together will help identify which are most complete and hence most suitable to start with for logical modelling. The remaining sentences may be referred to after an initial logical model has been produced to ensure that the logical model is complete and captures the intended meaning of the term of interest. This term of interest, the one we will logically model to capture its intended meaning, is referred to as the *defined term*. The defined term for this research is *CSAR operations*.

A sentence was found from the available US doctrine that provides a definition for the defined term:

[redacted]

Executive Summary, Page ix (Joint Chiefs of Staff 1998)

To create a logical model for this sentence the defined term is first identified and highlighted within the sentence using a bold font which is underlined with no italics. The rest of the sentence is placed into italics:

[redacted]

Starting Sentence

With the defined term and a sentence defining it identified the next stage of Logical Modelling is to list out the key words that give it meaning. The key words will be undefined and their meaning will be determined by relations between the words, which will be represented graphically, using the class diagram from the Unified Modelling Language (UML) (Object Management Group 2007). Modelling approaches like this have been used before, e.g. (Hatley, Hruschka and Pirbhai 2000, IEEE Computer Society 2000). The *key words* in the definition are identified and highlighted within the sentence using a bold font which is not underlined with italics:

[redacted]

Keywords in Starting Sentence - Sentence 1

Whilst not used at this stage note that other words in the definition would be shown in a bold font with no italics and additional words not in the definition would be shown in a smaller font with no bold nor italics. A matrix illustrating the use of font weights and accents for each word type within a logical model is shown in Table 28 – Logical Modelling Natural Language Notations in Appendix K, Logical Modelling .

With the defined term and key words identified a class diagram can be produced using the graphical notation that nouns are placed into boxes, verbs and relations are placed on lines and that solid boxes and lines are used for key words and other words from the definition, otherwise dot-dash graphics are used (C. Dickerson 2008). The initial Logical Model is shown in Figure 51.

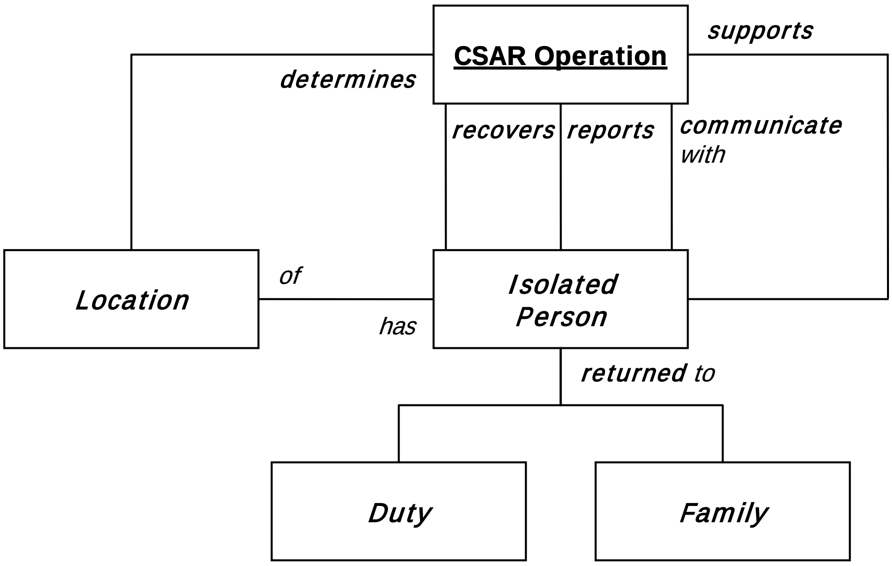


Figure 51 – First Logical Model of Combat Search and Rescue

The logical model presented in Figure 51 appears to be incomplete. The model shows that a CSAR Operation reports that the Person is Isolated but, logically, this seems incorrect. A CSAR Operation would be initiated by the report of an isolated person. This can be clarified through inclusion of supporting sentences. A search through the doctrine failed to find a suitable sentence containing the term of interest *report*. However, the literature refers to multiple terminology apparently as equivalents. For example:

[redacted]

Executive Summary, Page ix (Joint Chiefs of Staff 1998)

Comparing these five stages with those established in Starting Sentence we see that the terms *reporting* and *awareness and notification* appear to be equivalents. This equivalence will allow the transformation of sentences containing these equivalent terms into ones which refer to the context we are interested in: the role of reporting in a CSAR Operation.

[redacted]

Stages of a CSAR Operation, Page III-1 (Joint Chiefs of Staff 1998)

Using the above sentence and making the language of the sentence compatible with the defined term's sentence (substituting *CSAR Operation* for *rescue process*) the following sentence can be modelled:

<u>Notification</u> *of a downned aircraft or* ***isolated person begins*** *the rescue process [**CSAR Operation**].*

Sentence 2

The model of the preceding Sentence 2 produces the logical model shown in Figure 52.

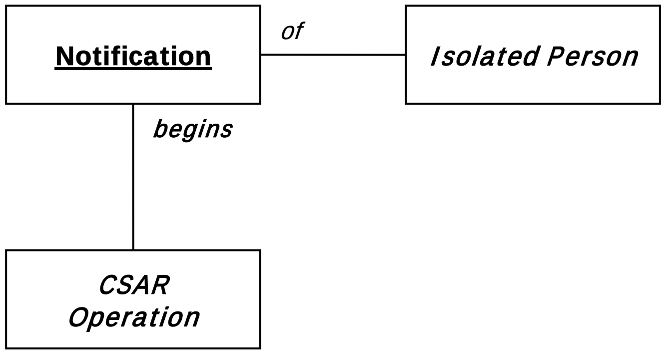


Figure 52 – Logical Model of Notification

This is still incomplete as the model does not capture what or whom does the notification which begins the CSAR Operation. Another supporting sentence provides this information:

[redacted]

Stages of a CSAR Operation, Page III-2 (Joint Chiefs of Staff 1998)

This sentence has been chosen as a supporting sentence as whilst no sentence directly links notification back into the structure of the defined term's sentence, this sentence does imply notification through the phrase *forward the details* which from the surface structure of the sentence refers to *information about isolated personnel*.

[redacted]

Sentence 3

Sentence 3 is modelled in Figure 53 below. Note the implied relationship shown that *notification contains information about isolated personnel*.

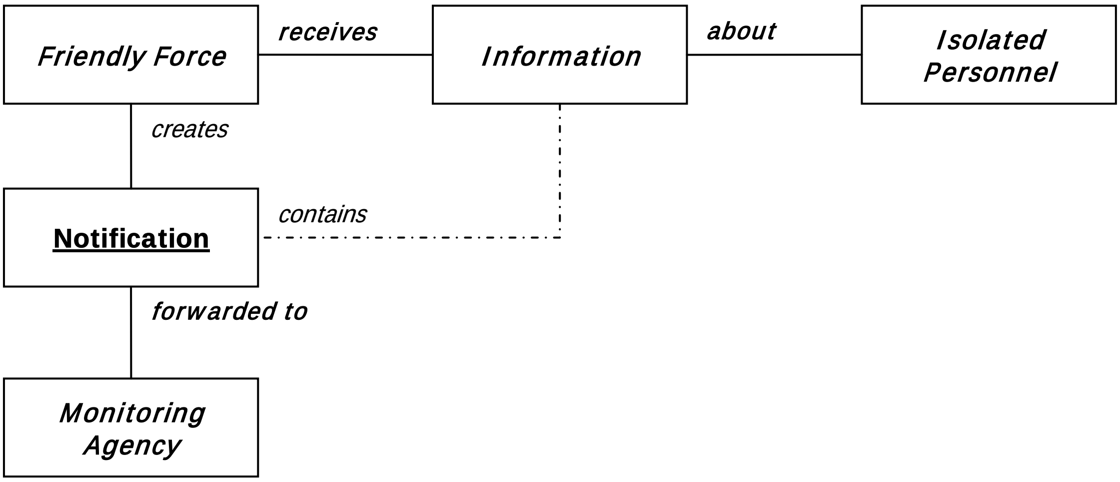


Figure 53 – Identifying the Source of Notification

The interpretation of notification as details of information about isolated personnel is further reinforced by another supporting sentence:

[redacted]

Stages of a CSAR Operation, Page III-1 (Joint Chiefs of Staff 1998)

This sentence provides information of the ultimate source of the notification information, that of the *Joint Search and Rescue Centre*. This is a specific US facility that plays a pivotal role in US Combat Search and Rescue doctrines (Joint Chiefs of Staff 1998) (Joint Chiefs of Staff 1996) (United States Air Force 1998). For the purposes of this research a more generic, non-specific organisation will be referred to as the *Combat Search and Rescue (CSAR) Organisation*. This phrase is substituted into the logical model presented in Figure 53 in the updated logical model below, Figure 54.

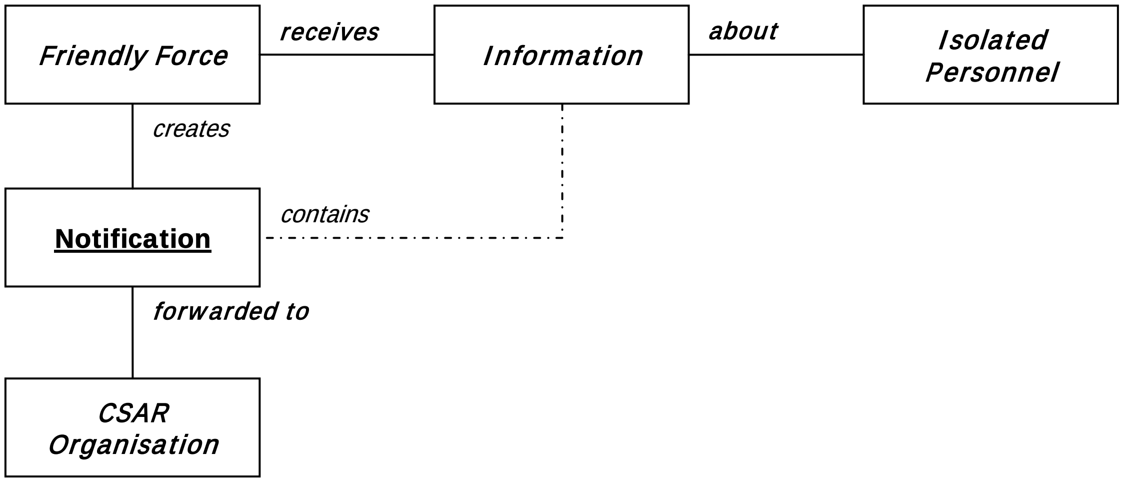


Figure 54 – Generalising the Notification Recipient

Whilst Figure 54 captures the source of the notification it implies that it is a passive process without involvement from the *isolated personnel*. This can be clarified through another supporting sentence:

[redacted]

Stages of a CSAR Operation, Page III-1 (Joint Chiefs of Staff 1998)

Using the above sentence the following sentence can be modelled:

[redacted]

Sentence 4

Note that at this level of logical modelling the specifics are not important and hence the multiple platforms that an isolated personnel might contact (*wingman*, *escort aircraft*, etc.) have been captured through the most generic phrase in the sentence, *friendly forces*, which allows us to integrate this information into the logical model shown in Figure 54. The updated logical model is shown in Figure 55.

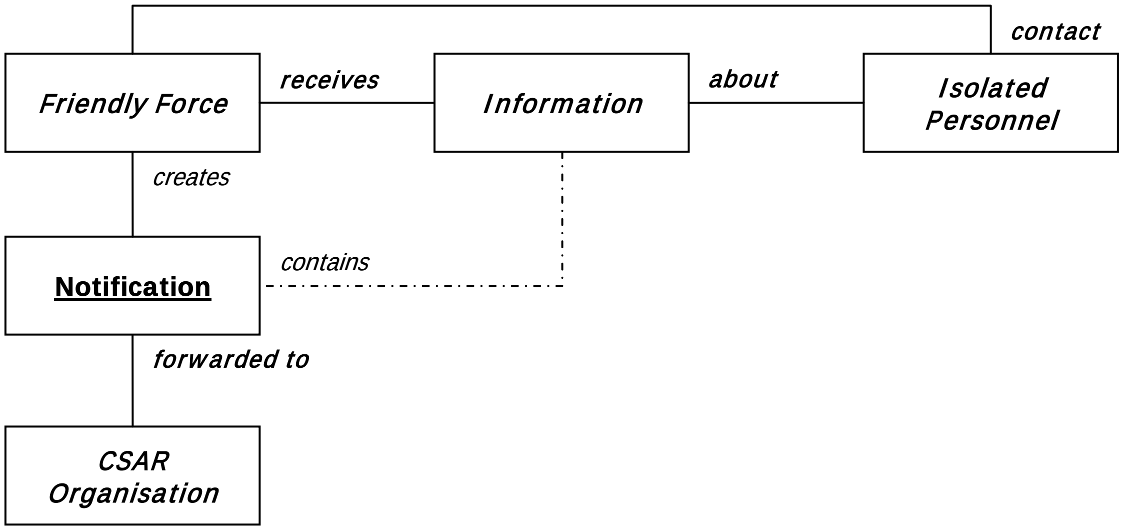


Figure 55 – Establishing Isolated Personnel Relationships

The logical model shown in Figure 55 appears to be complete and so it can be integrated with Figure 52 – Logical Model of Notification, as shown below in Figure 56.

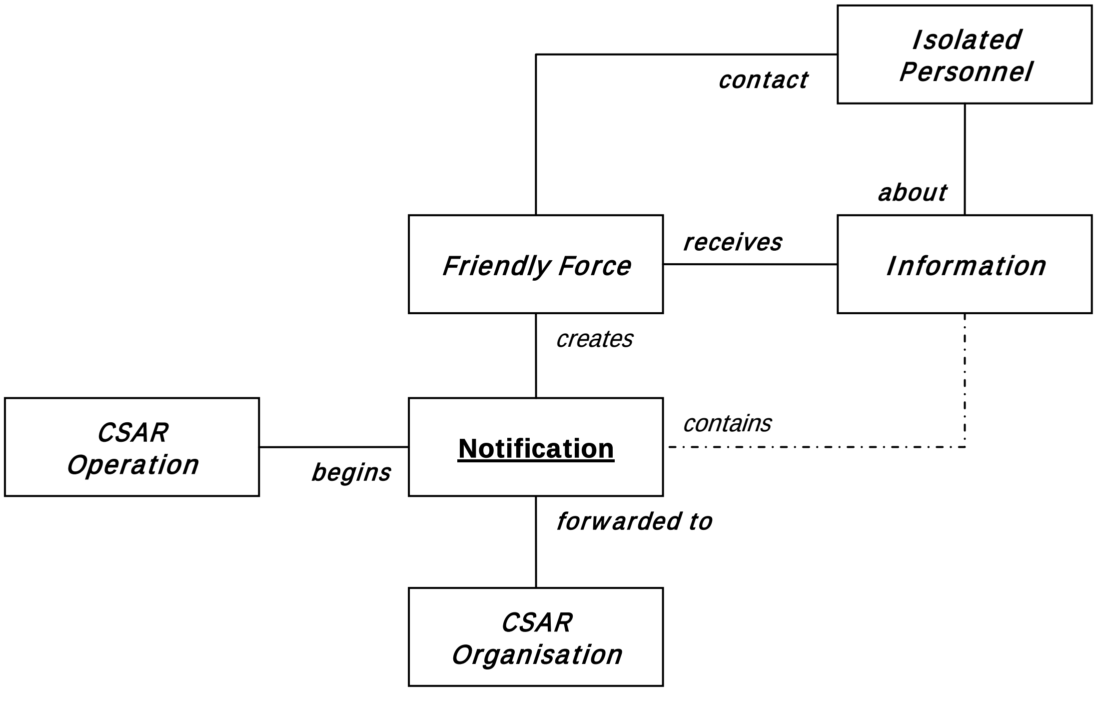


Figure 56 – Integrating a Logical Model of Notification

With the notification logical model established as shown in Figure 56 this was integrated with the First Logical Model of Combat Search and Rescue shown previously in Figure 51. The integrated model is shown in Figure 57 and addresses the original issue that the CSAR operation reported an isolated person. Through the logical exercise the updated reporting model and its relationships with the CSAR system have been established.

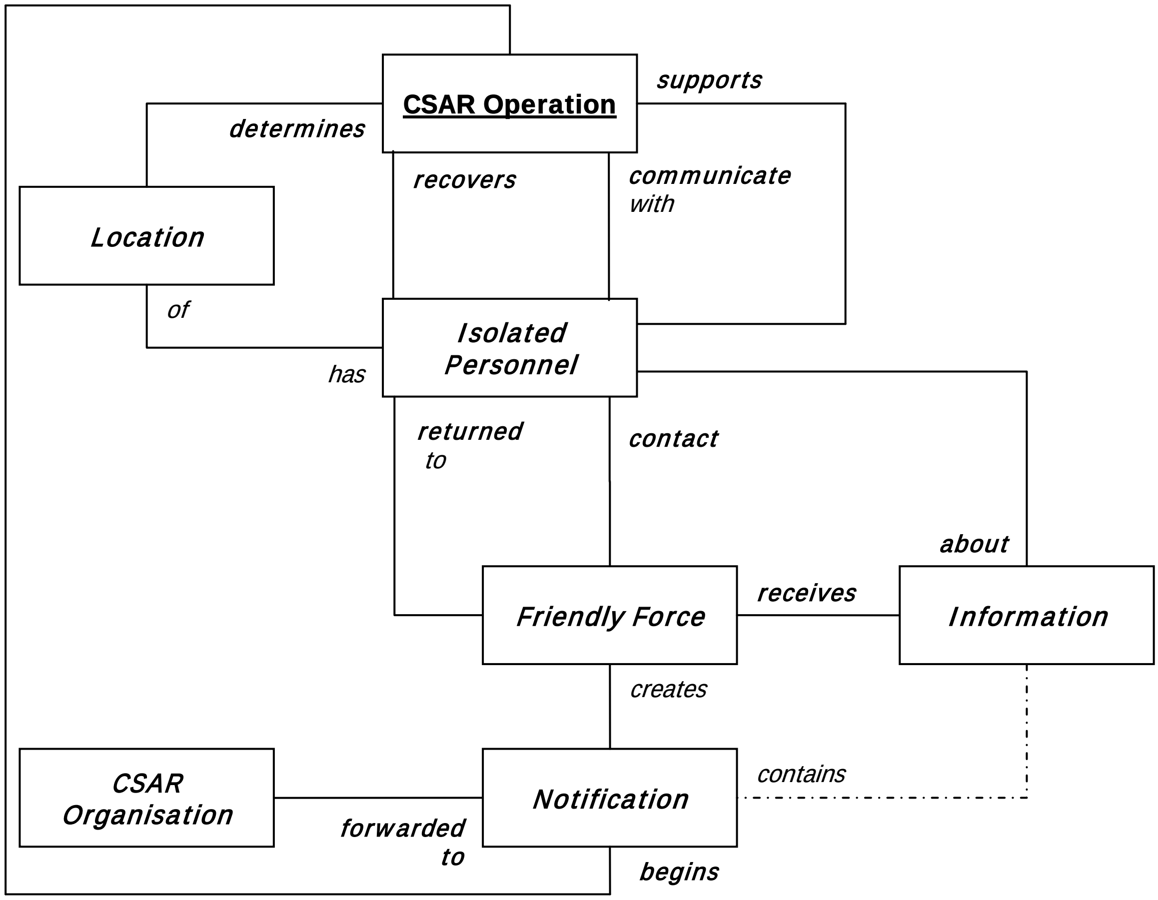


Figure 57 – Second Logical Model of Combat Search and Rescue

The updated logical model of CSAR (shown in Figure 57) is now complete and incorporates the original CSAR logical model with the extended reporting/notification logical model. However, the logical model for CSAR contains unnecessary details for the current purpose of the architecture. In the original CSAR logical model we sought to clarify the *report* relationship. To achieve this required an in depth consideration through further logical modelling to understand the underlying concepts and relationships. With these established and captured we can now reduce the logical model of the defined term *CSAR Operations* by bounding the extended logical model of notification, as shown in Figure 56, as the term report from the original logical model. Recall that the terms *reporting* and *awareness and notification* have been established as equivalents. The Report function's logical model can now be separated from the CSAR logical model and is shown in generic form in Figure 58.

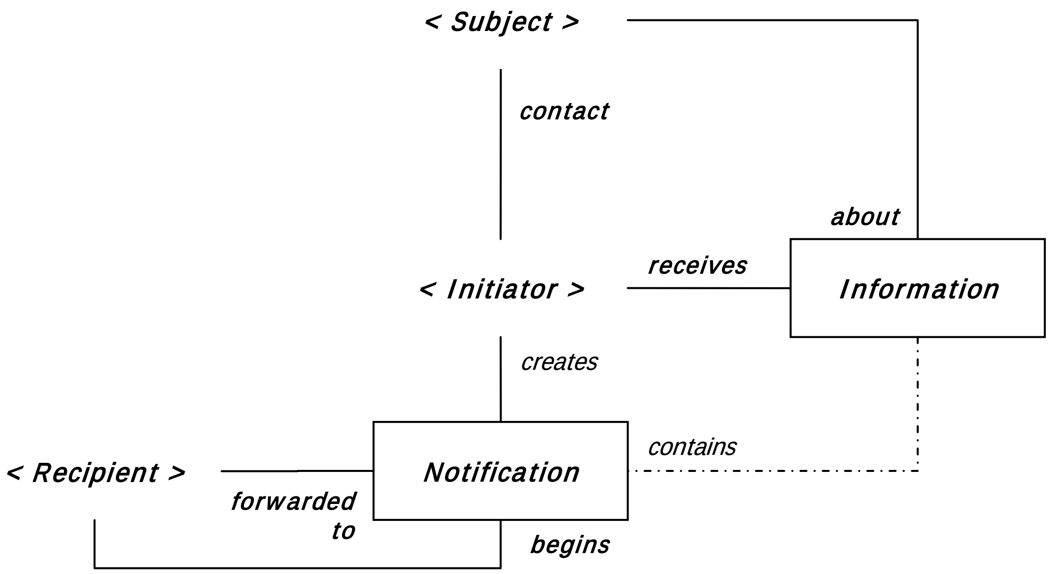


Figure 58 – A Logical Model for Report

With the *report* relationship substituted for its logical model a final logical model for CSAR can be produced and is shown in Figure 59.

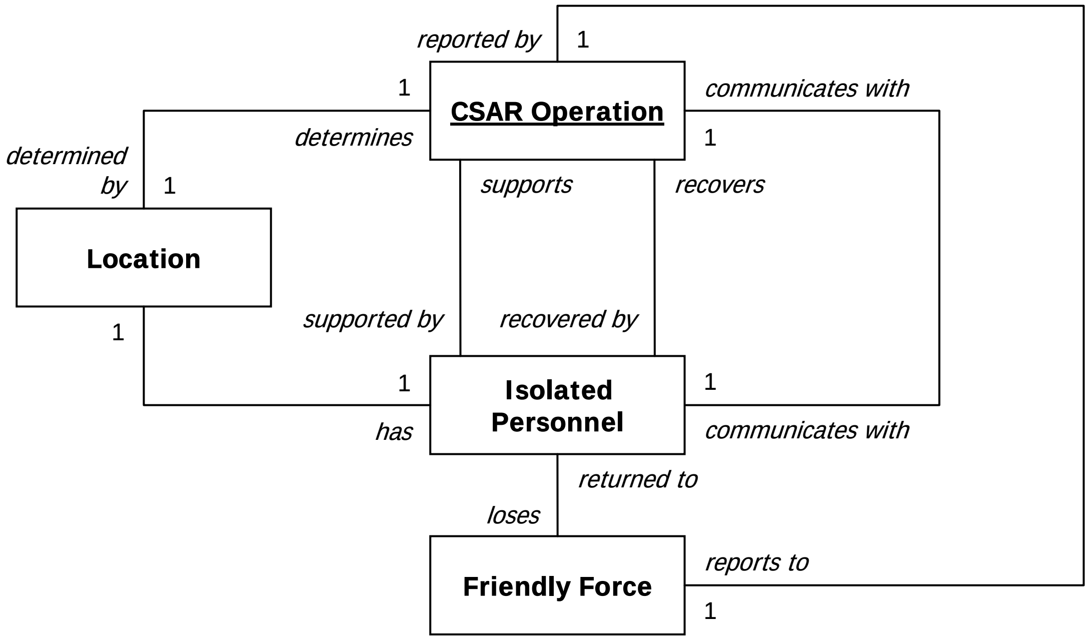


Figure 59 – Final Logical Model of Combat Search and Rescue

The final logical model for CSAR shown in Figure 59 captures the essential model of CSAR missions. For the software tool we need to understand the processes and flows within this model. To do this we need to transform this logical model into a flow diagram, which is the focus of the next subchapter.

6.4 Logical Model to Flow Diagram Transform

For the logical model established in the previous subchapter to be of utility in the second version of the DSS software tool we need to represent it in terms of processes and flows. This process moves from the specific problem context to a domain independent viewpoint of the system which forms the CIM. This ultimately allows greater flexibility as instead of considering context specific problem spaces we move towards a higher level representation which, beyond the scope of this research, can be used to match and group problems to seek common solutions. The benefits of this can be further explored through Executable Architectures allowing translation of disparate problem contexts with common solution models. This is explored in later in this thesis in Chapter 8, subsection 8.4.

The process discussed in this section is the transformation approach used for this research. There are different ways of creating process and flow diagrams. The process and flow diagram must be the end result of whatever approach is used as the Integrated Model Driven Approach requires this as an input (see Figure 46). Hence, the approach presented is an exemplar. For the purposes of this research it doesn't matter whether the output of this exemplar is right or wrong as the Integrated Model Driven Approach is the constant being tested. In practise the resultant process and flow diagram would be reviewed by appropriate subject matter experts and iteratively maintained to ensure an accurate and timely representation of CSAR missions.

The first stage to transform the logical model to a flow diagram is to recognise that each relation in the logical model is a process. These processes are identified and captured as a bubble on the relationship line and the relationship words are removed, as shown in Figure 60.

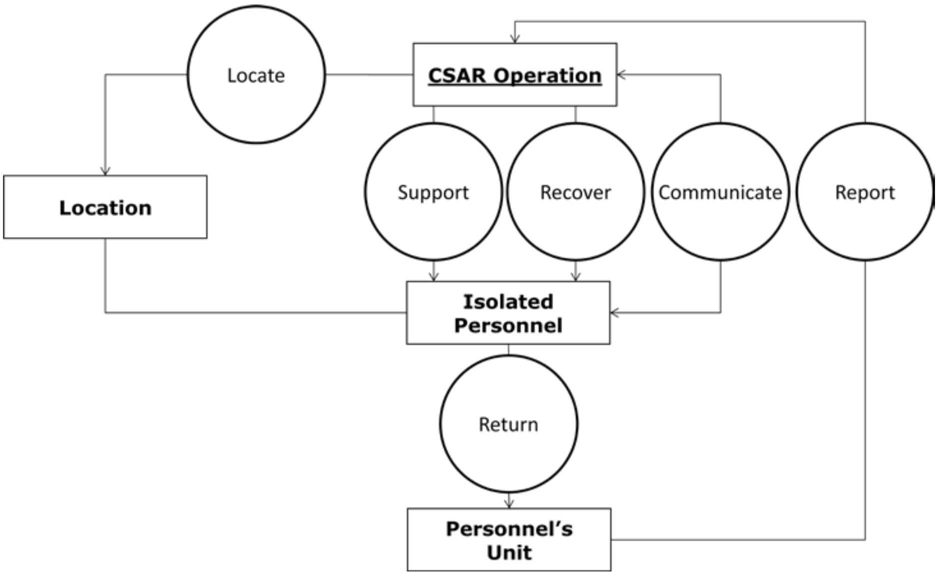


Figure 60 – Identifying Processes within the Logical Model

With the processes identified the context can be removed from the diagram. While the sequence of keywords in Figure 60 sets precedence of the processes there is no indication of precedence between keywords. For example it can be seen that *Report* occurs before *Locate*, but the precedence between *Locate*, *Support*, *Recover*, and *Communicate* is not clear. An interpretation is required where multiple processes occur between keywords and an assessment made of precedence. The interpreted sequence is shown in the process diagram shown in Figure 61.

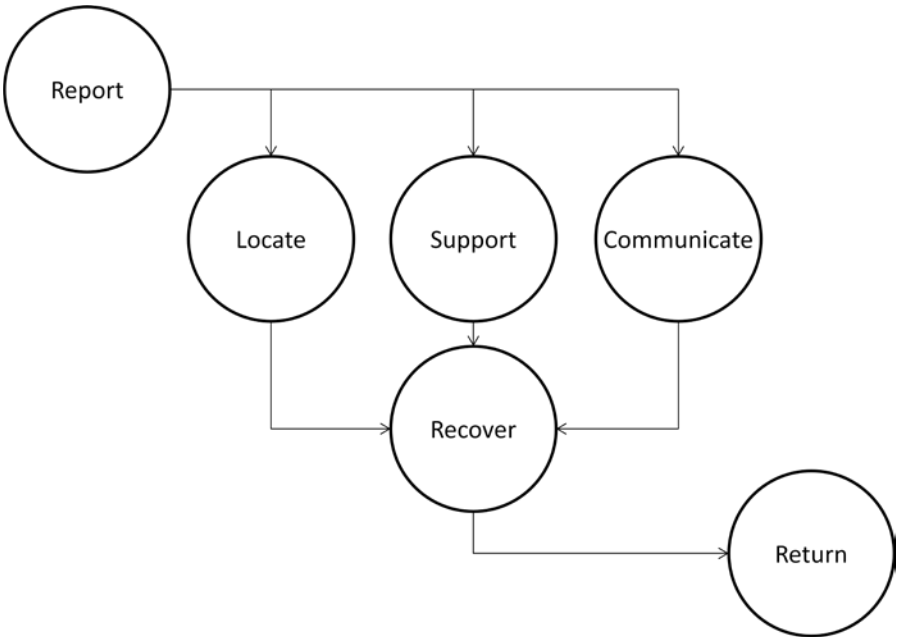


Figure 61 – A Process Model of CSAR Operations

The processes identified in Figure 61 can be given precise definitions to support future activities:

Report – the process of reporting the isolated personnel as requiring CSAR.

Locate – the process of locating the position of the isolated personnel.

Support – the process of supporting the isolated personnel with information.

Communicate – the process of communicating with the isolated personnel.

Recover – the process of physically recovering the isolated personnel.

Return – the process of returning the isolated personnel to their unit and families.

With the process diagram captured the final stage of the transformation is to identify flow between the processes. This is achieved through interpretation of the original logical model to identify the subject of the flows and the type of flow. The completed flow diagram for CSAR operations is shown in Figure 62.

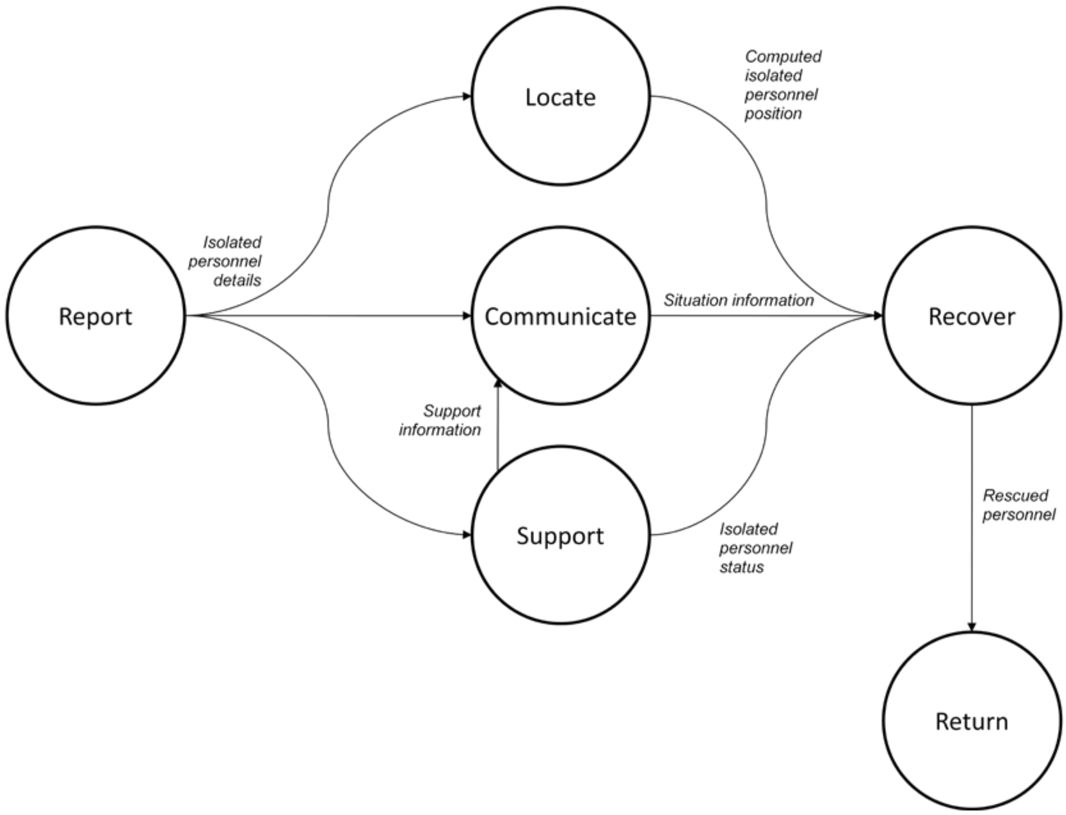


Figure 62 – A Flow Diagram for CSAR Operations

The model shown in Figure 62 leads to a realisation that in the definition of CSAR missions the element the decision maker is concerned with is not the entire model, rather it is a sub model incorporating the processes between the starting Report and ending Return processes, but not including them. The Report process must be done to initiate the mission and hence the need for the military decision maker to decide what SoS to employ for the mission. The Return process occurs once the personnel has been safely recovered and hence is beyond the scope of the decision makers consideration as it does not impact the SoS to be used. As defined this boundary is illustrated in Figure 63.

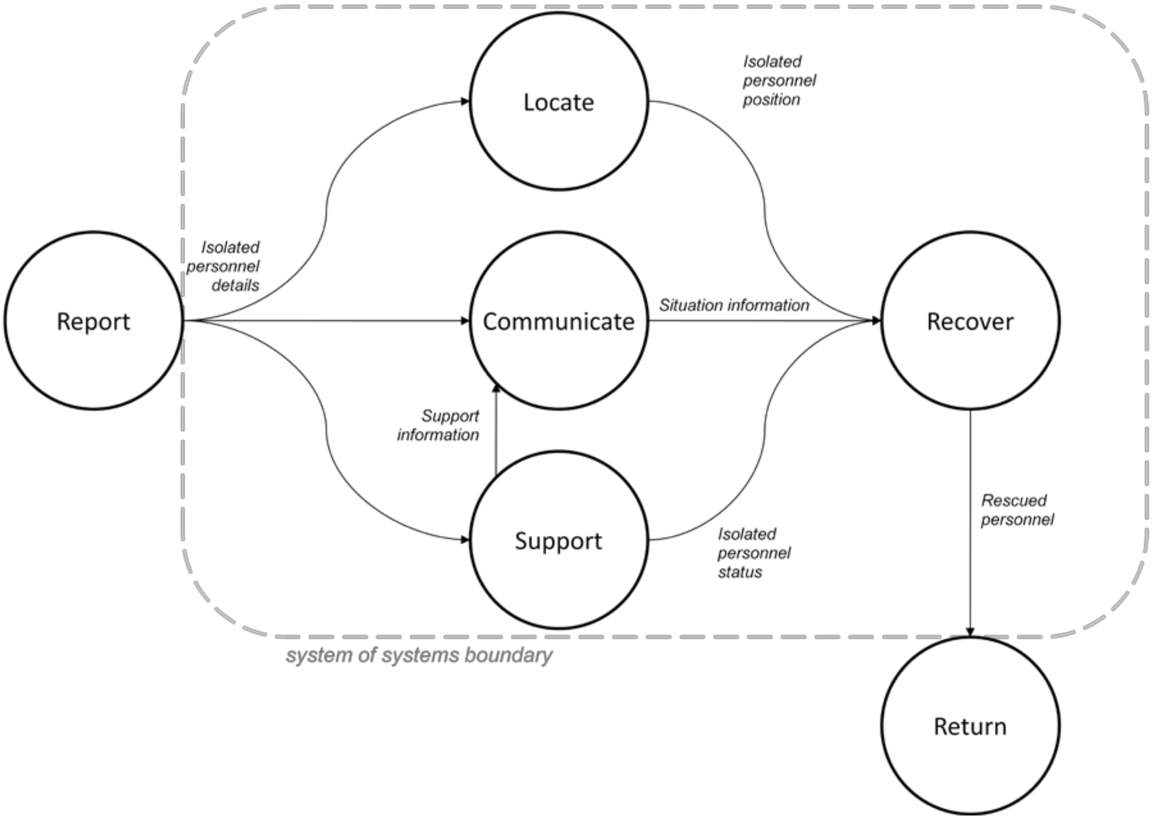


Figure 63 – SoS Boundary for CSAR Operations

With the flow diagram established for CSAR operations this can now be captured in the developed DSS software tool, which is the subject of the next subsection.

6.5 Determining the CIM and PIM for CSAR

The flow diagram established in the previous section contains the complete set of information necessary to generate the CIM and PIM and hence necessarily contains the information for the CIM-PIM transform. The CIM captures the processes and their relationships and the PIM captures the flows and their relationships. This is illustrated in the example shown in Figure 64 (note the relationships shown are illustrative and should not be confused with the relationships shown previously in Figure 63).

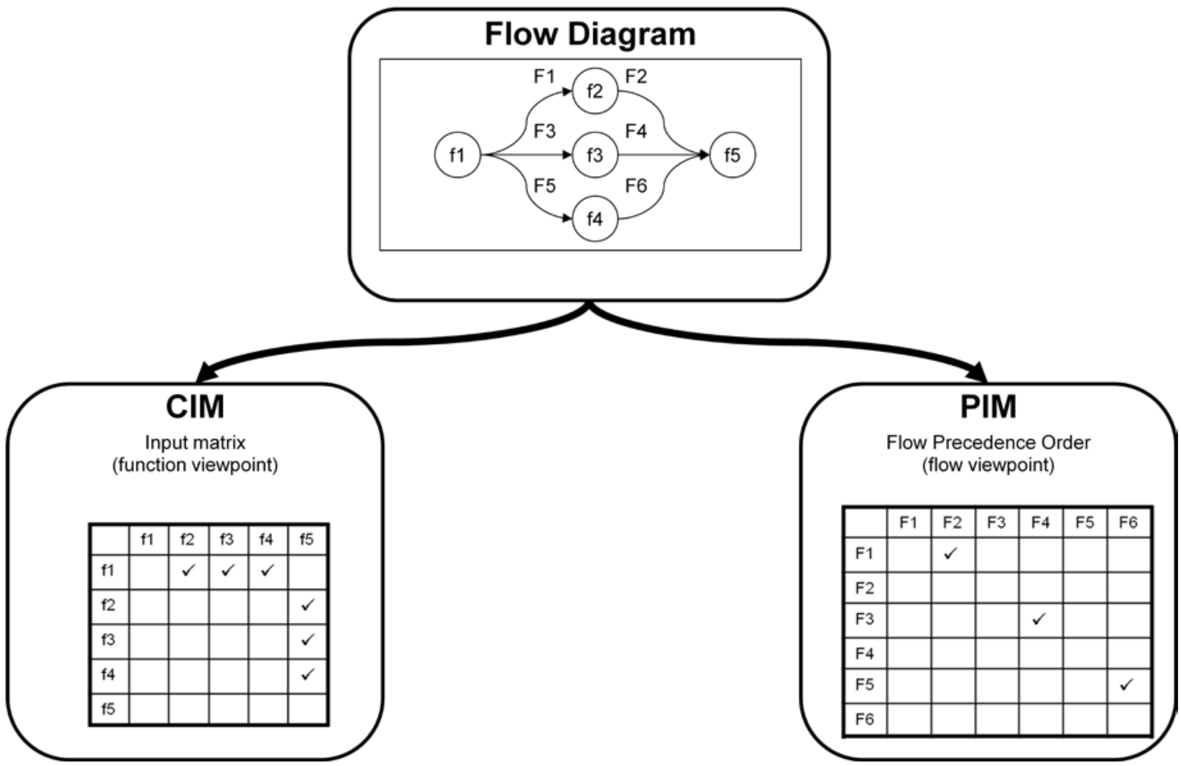


Figure 64 – The Role of the Flow Diagram to Determine the CIM and PIM

Note how the flow diagram, contains the complete set of information for the CIM, PIM and the transformation between these models as described previously in section 5.5. The CIM contains the processes whilst the PIM contains the flows. The established process relationships for the CIM are shown in Matrix 4. As with the relationship matrices presented in previous Chapters this is read row/column, so for example the *Report* process is related to *Locate*, *Communicate* and *Support*.

Matrix 4 – CIM Relationship Matrix for CSAR

	Report	Locate	Communicate	Support	Recover	Return
Report		✓	✓	✓		
Locate					✓	
Communicate					✓	
Support			✓		✓	
Recover						✓
Return						

The CIM only captures part of the established Flow Diagram for CSAR as shown previously in Figure 62. The other information in Figure 62 are the flows between the processes. The relationships have been defined by the flow sequences shown in Figure 62. The PIM relationships are show in Matrix 5.

Matrix 5 – PIM Relationship Matrix for CSAR

	Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
Isolated personnel details		✓	✓	✓	✓	
Situation information						✓
Isolated personnel position						✓
Isolated personnel status						✓
Support information		✓				
Rescued personnel						

This provides a generic model for CSAR operations which can now be utilised to conduct a repeat set of historical case studies in the next chapter.

6.6 Methodology for System Profiles

Two profiles are required by this process – a profile of the systems and a profile of the systems as utilised by the commander (expressed as commanders intent). In either case to profile the systems they were considered as "black boxes" and the flows crossing the determined system boundary were identified, as illustrated in Figure 65. This explicit bounding of the system and consideration of only the flows makes assessment far easier as the internals of the system are no longer of interest, only those which penetrate the boundary.

Figure 65 – Bounding a System to Establish Flows

6.7 The Evaluation Process

With a CIM and PIM captured the tool is ready to evaluate CSAR operations. The overall methodology for this is shown in Figure 66 and considers two viewpoints on the system – the *mission viewpoint* and the *commanders viewpoint*.

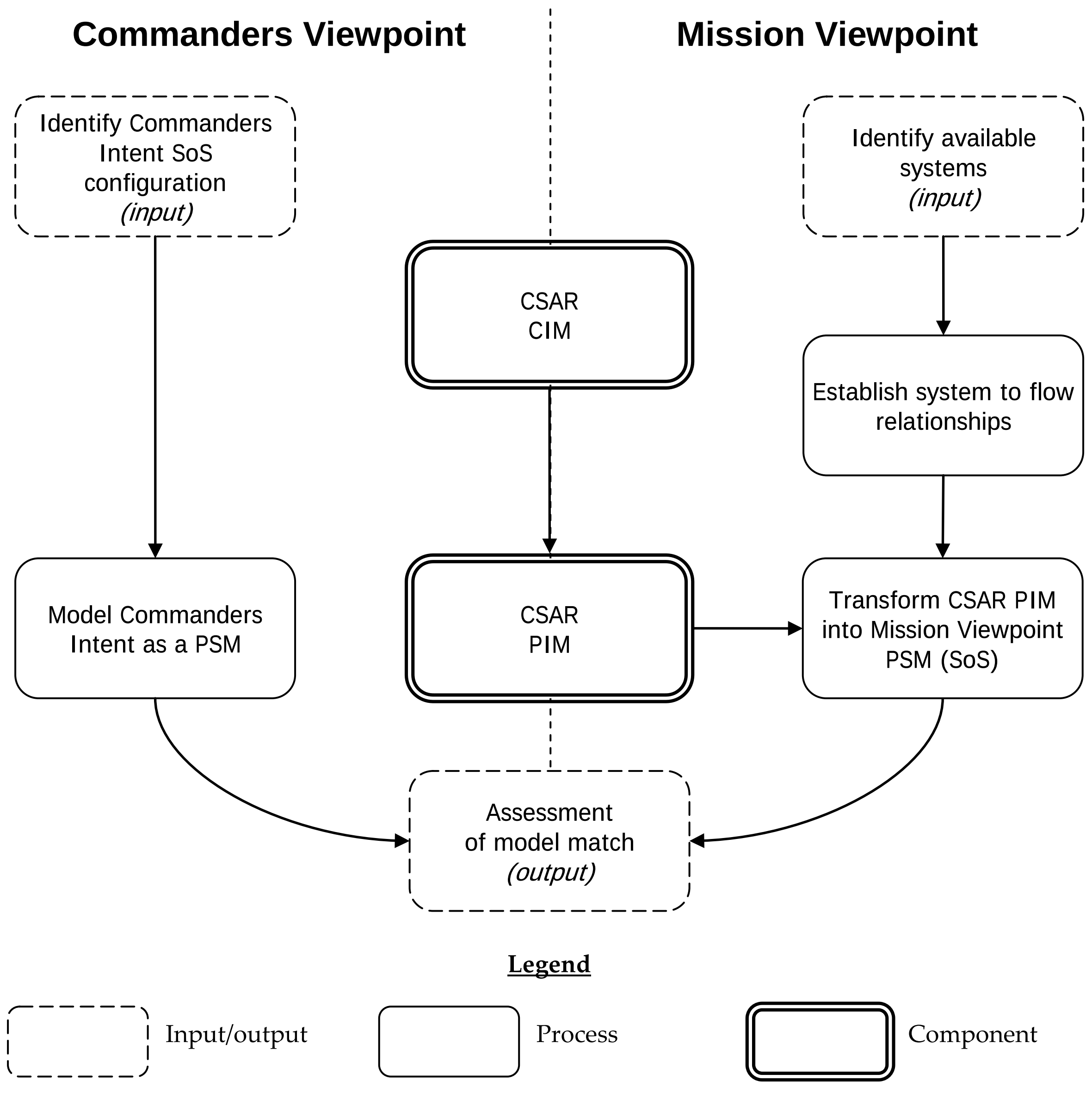


Figure 66 – An Architecture of a Model Driven Approach for a DSS Utilising CIM, PIM, PSM

The mission viewpoint represents the mission's requirements as established through logical modelling and the transformation of the logical model into a flow diagram which serves as an input into the CIM. This is used as the baseline of what is needed to conduct the mission as expressed as processes (in the CIM) and flows (derived in the PIM). The mission viewpoint can be seen in the second set of case studies presented in the next chapter in section 7.2.1 for Case Study One and section 7.3.1 for Case Study Two.

The Commanders Viewpoint captures the commanders intended SoS configuration to conduct the mission. This SoS is constructed by the commander as described by the RPDM within a conceived operational procedure. The Commanders Viewpoint can be seen in the second set of case studies presented in the next chapter in section 7.2.2 for Case Study One and section 7.3.2 for Case Study Two. The DSS provides support to the decision maker on their selection and configuration of the SoS based on an procedure independent model. This support is based on the *process & flow* perspective which approximates to the functional perspective found in the original DSS. To clarify, the DSS does not consider how the SoS and the individual systems within it are utilised to conduct the mission in terms of operational conduct. Rather, the DSS serves the decision maker by comparing the selected systems and the inter-system relationships defined to form an SoS to an implementation independent model of CSAR missions. This model is independent of both technology (i.e. hardware) and conduct (the devised plan for the mission). To expand on the concept of a conduct independent model consider a military mission. The commander will select a range of systems to form an SoS to conduct a mission (expressed as the commanders intent). The conduct will specify tactics, techniques and procedures, a plan of how the commander envisions the mission to play out. This conduct will always implement the same basic processes and flows and these have been captured as an implementation independent model in the Mission Viewpoint. The Commanders Viewpoint is captured at the SoS level in a PSM and system to flow relationships are captured as a PSM/PIM transform matrix. By completion of these two models the PSM can be transformed to a PIM. This allows comparison of the Mission Viewpoint to the Commanders Viewpoint at the PSM level.

The PSM shows the structure of the interconnection of the systems comprising the SoS. The Commanders Viewpoint shows the intended configuration of the systems and, by visualising the relationships, helps inform the decision maker of the structure of the SoS. For example, such visualisation helps show whether the system is distributed with a number of mutually interconnected systems, or centralised with a single system or clusters of systems connecting to the other systems in the SoS. The Mission Viewpoint PSM shows the recommended configuration of the SoS based on all possible interconnections of the systems. By comparing the two, the decision maker can identify options that they may have not considered to utilise. The comparison of the PSM can be seen in the second set of case studies presented in the next chapter in section 7.2.3 for Case Study One and section 7.3.3 for Case Study Two.

The next section considers how this approach supports the Recognition Primed Decision (RPD) model.

6.8 Supporting Recognition Primed Decision Making

The initial architecture of an approach for a DSS developed and implemented in the first prototype software tool supported Recognition Primed Decision (RPD) making through trying to provide a compatible support approach. The model driven approach implemented in this chapter provides an opportunity for closer integration with the RPD. The implemented approach recognises that with a decision maker using RPD and a mission requirement captured separately there will be two viewpoint at different ends of the MDA model spectrum. The mission viewpoint will produce a CIM/PIM while the RPD will produce a PIM. This leaves the question of how these two views can be reconciled. The methodology developed allows the two viewpoints to be compared and assessed at the same model level as shown in Figure 67.

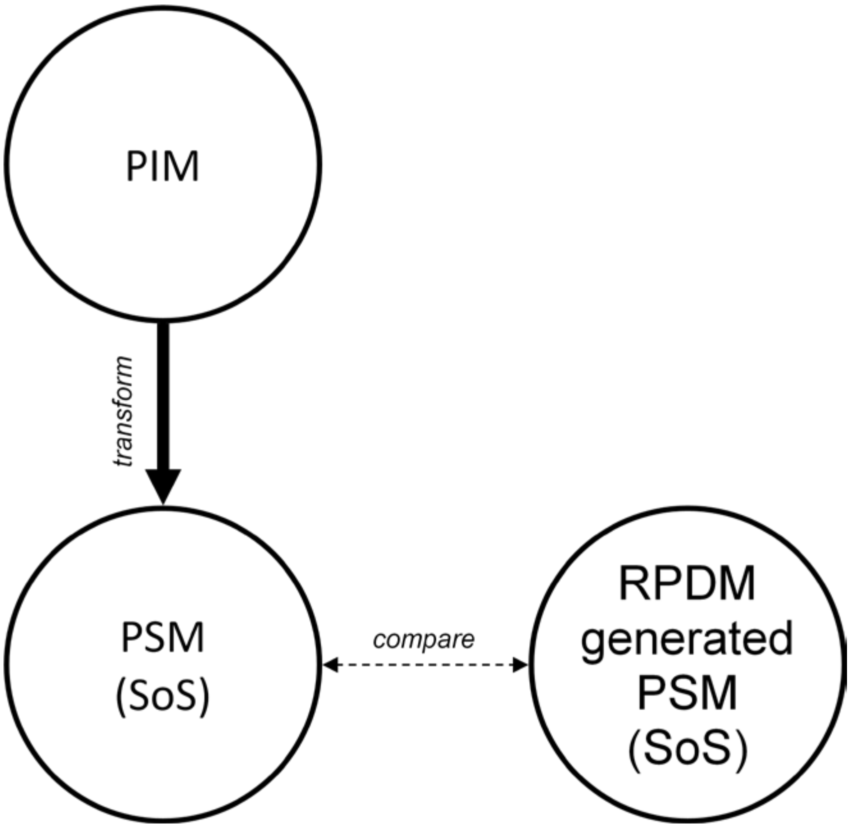


Figure 67 – DSS Tool Evaluation Process

The next chapter repeats the case studies originally conducted with the first prototype DSS software tool with the new DSS software tool. The transform shown in Figure 67 will be applied to the second set of case studies in the next Chapter 7, for the first case study in Matrix 8 and for the second case study in Matrix 18. The comparison in Figure 67 will similarly be shown in Matrix 15 for case study one and Matrix 22 for the second case study.

6.9 Chapter Summary

This chapter has outlined the development of the second implementation of an approach for a decision support system. The development of the underlying software tool was detailed, particularly the implementation of model transformations. As this approach required precise models the Generic Functional Model (GFM) developed for the first software tool was not appropriate. A logical model of the Combat Search And Rescue (CSAR) mission was created from available doctrine which required multiple sentences and models of these sentences to be combined into a single model. With a logical model for CSAR missions established this was then converted into a flow diagram. The flow diagram in turn could be used to determine a Computational Independent Model (CIM) and Platform Independent Model (PIM) for CSAR missions. These are the two main components for the model driven approach developed in the previous Chapter 5. The methodology for profiling systems was presented and finally the overall evaluation process. With the full approach outlined and a software tool developed the next Chapter 7 evaluates the developed tool with a repeat of the case studies conducted with the original parametric approach.

Chapter 7

Evaluation of Second Approach for a Decision Support System

Outline of Chapter

This chapter revisits the case studies conducted earlier in this thesis using the new model driven approach developed in Chapter 5 and implemented in a software tool in Chapter 6. The two case studies originally conducted in Chapter 4 with the original parametric approach are repeated. For each of the case studies a Mission Profile and Commanders Intent Profile is generated. These profiles are compared and conclusions drawn. Finally, in light of the revisited case studies an overall evaluation of the tool and the model driven approach is given.

Research Contributions of Chapter

From the previous chapter(s):

*This **Research** established a **Problem** and **Objectives** to develop an **Architecture** of a **Model Driven Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics** which informs the decision maker.*

*In the model driven approach the system of systems is captured in a **Platform Specific Model**, the common characteristics are captured in a **Platform Independent Model** and the combat search and rescue mission is captured in a **Computational Independent Model**. These models can be related by **Model Transformation**. The architecture has been implemented in a **Software Tool**.*

This chapter introduces:

*The software tool was used to conduct a set of literature based **Case Studies** of the application of this research (combat search and rescue) and the results evaluated.*

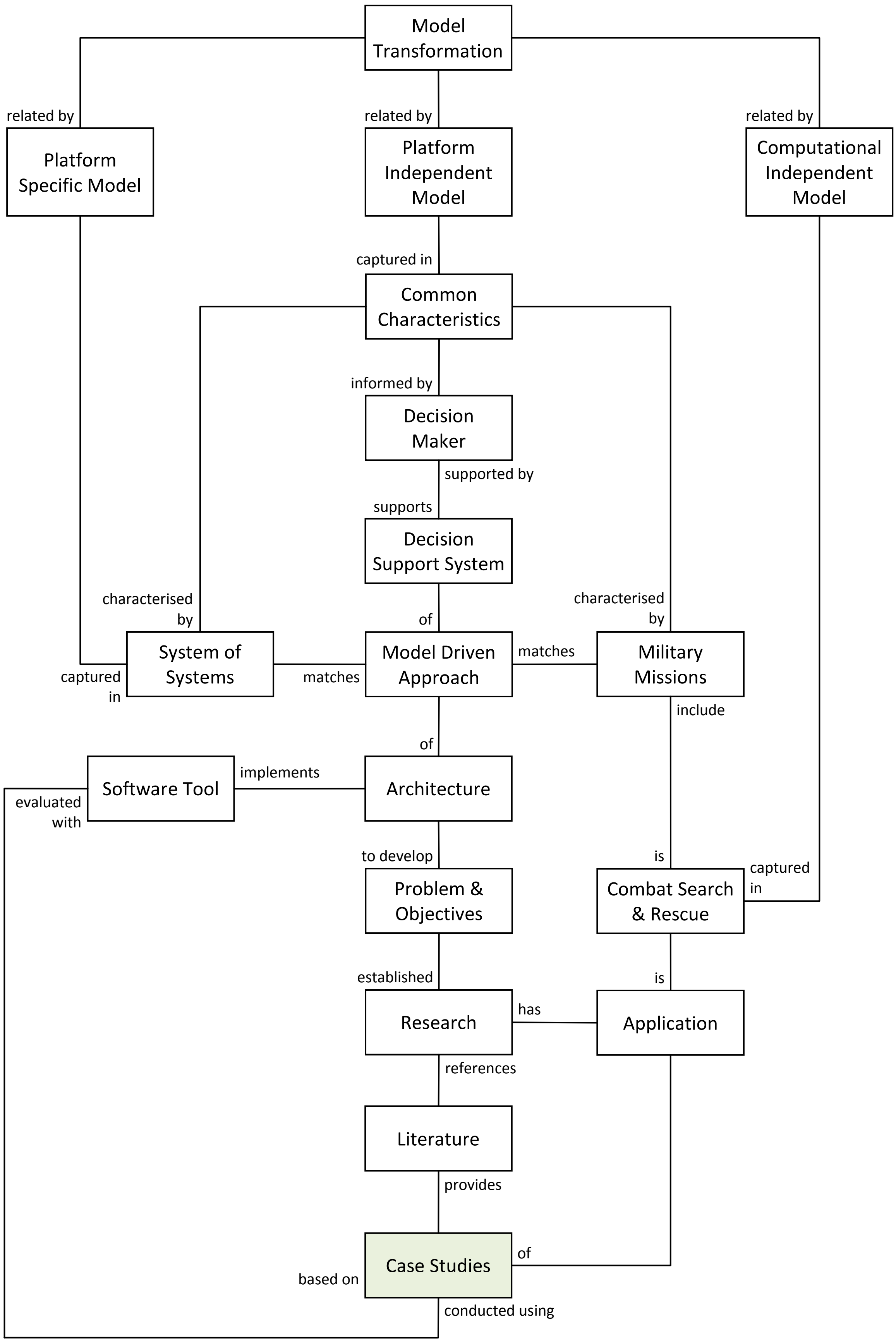


Figure 68 – Logical Model of Research Contributions of Chapter Seven

7.1 Revisiting the Case Studies

This section reconsiders the case studies conducted in Chapter 4 with the methodology developed in the previous Chapter 6. The methodology adds rigour and precision to the techniques developed in the first part of this research to profile missions and systems using a common set of characteristics. The previous chapter used logical modelling to establish a precise model of CSAR operations which was an evolution of the Generic Functional Model (GFM) developed earlier in this research. The logical model was used to determine the Computational Independent Model (CIM) and Platform Independent Model (PIM) for CSAR Operations. The creation of these models enabled a Model Driven Architecture (MDA) approach to be used for the Decision Support System (DSS). The software tool implementation of the DSS was rewritten to support these enhancements. In this chapter the original case studies will be repeated with this new enhanced methodology and the findings analysed against the historical evidence and also compared and contrasted with the findings from the first DSS software tool.

As with the first implementation three main aims were set to be achieved by developing this second prototype software tool:

1. **Feasibility** - demonstrate that the developed DSS architecture could be implemented in a software tool.
2. **Validation** - perform a set of historical case studies utilising the developed prototype software tool.
3. **Guide** - guide the further development of the architecture of an approach for a DSS.

These aims will be addressed the evaluation of the second implementation in section 7.4.

7.1.1 Generated Mission to System Profile

With the CIM and PIM profiles for CSAR operations established in the previous chapter the DSS tool can be used to profile the mission and systems and transform the established PIM to a specific mission PSM. An overview of the PIM to PSM model transform and comparison with the Commanders Viewpoint (the RPDM generated PSM) is outlined below in Figure 69. The Commanders Viewpoint is the PSM generated by the Recognition Primed Decision model as identified as the most representative model of how military decision makers actually make decisions as discussed previously in section 2.4.

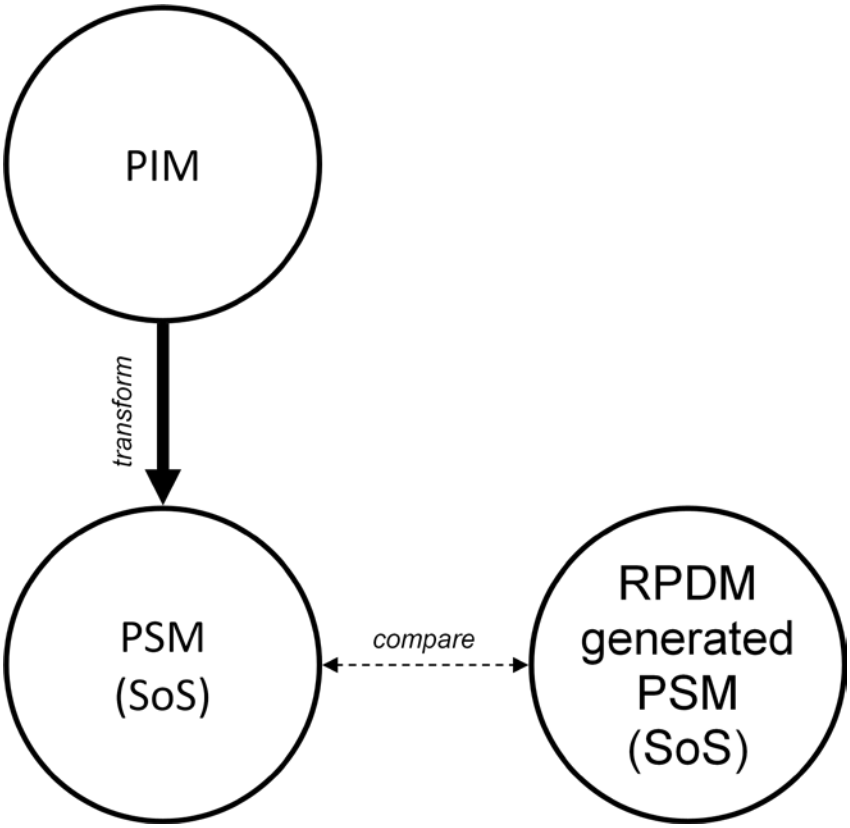


Figure 69 – Model Transformation Sequence

The rest of this chapter reconsiders the case studies conducted in Chapter 4 with the methodology developed in the previous Chapter 6.

7.2 Case Study One Revisited – Seawolf CSAR in Vietnam

This subsection revisits the case study previously conducted in subsection 4.3.1, Case Study One – Seawolf CSAR in Vietnam. The narrative for this case study was previously given in subsection 4.3.1.1. This case study was repeated using the methodology shown previously in Chapter 6 and sought to understand the implications of the various SoS configurations against mission requirements. The four systems profiled for this case study are:

- A rescue configuration Huey helicopter
- An attack configuration Huey helicopter
- Air Force Controller
- 57th Medical Attachment Controllers

Consider the developed flow diagram for CSAR operations developed in section 6.4 and shown again in Figure 70.

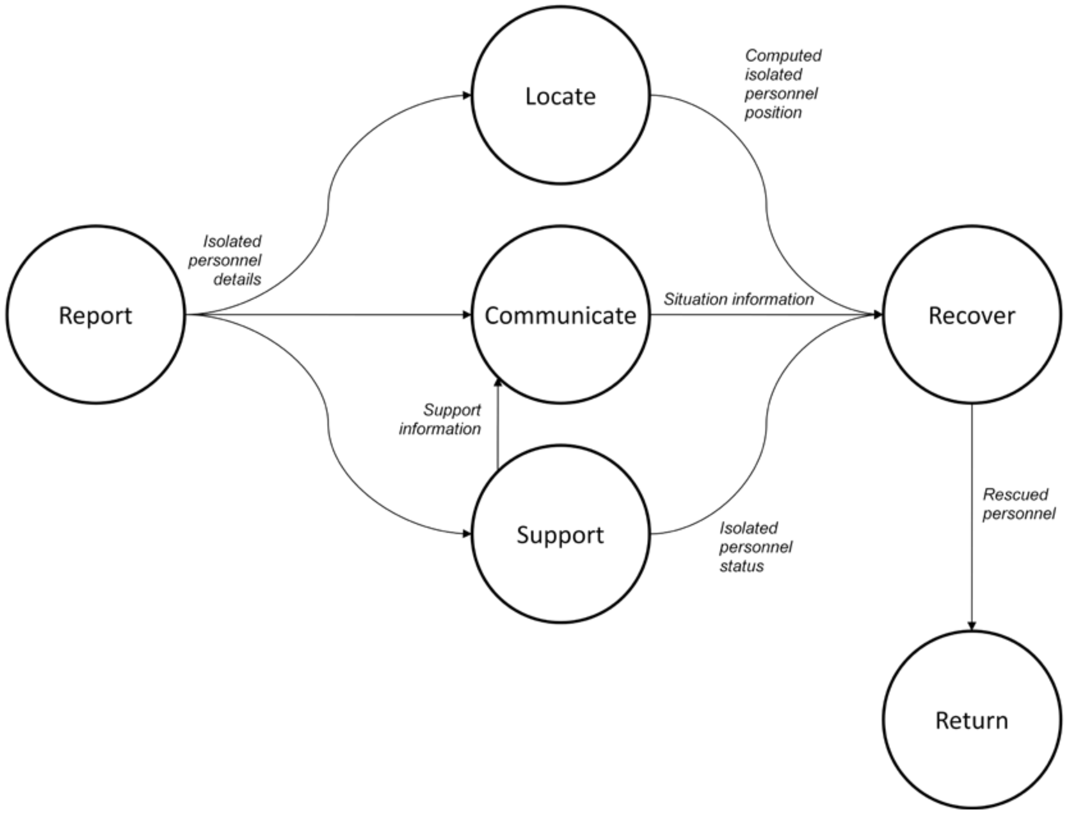


Figure 70 – Developed Flow Diagram for CSAR Operations

This developed flow diagram for CSAR operations is a template for conducting CSAR operations and modified diagrams will be developed for this case study based on the roles and interactions of the systems available in this case study. There are two viewpoints which are modelled by the DSS. The first is the mission viewpoint of the system profiles which captures the possible flows and relationships between the systems. The second viewpoint is that of the commander's intent which captures the systems profile as the decision maker wants to utilise them. The following two sections determine profiles for these two viewpoints, the mission and commanders intent respectively.

7.2.1 Mission Profile

The purpose of this subsection is to profile all the possible connections between the available systems which are modelled as *flows*. When the mission has been profiled and transformed into an implementation independent model the profiles established in this section will provide the transformation into the implementation. The purpose of this is twofold: firstly to highlight the possible interconnections between systems that the decision maker is not currently utilising (SoS awareness) and secondly to provide a mapping of the systems to flows to allow multiple SoS configurations to be assessed. This mapping can be used when comparing the mission generated SoS to the decision makers SoS to trace capability gaps back to the SoS relationships. The flows were determined by considering available references. In actual use the profile would be completed by subject matter experts to accurately reflect the system to flow relationships. The full set of determined possible connections is illustrated in Figure 71.

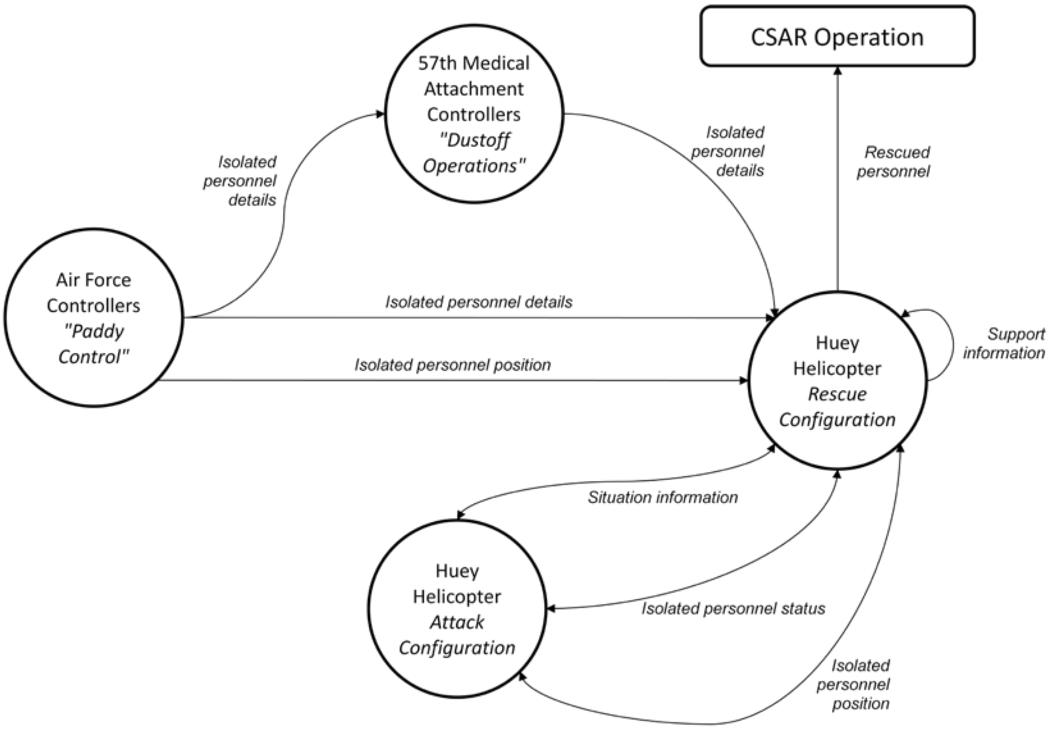


Figure 71 – Possible Flows between Systems

The system to flow relationships can be determined from Figure 71 and are captured in Matrix 6. These relationships were determined by the author based on available technical documentation about these systems and the role they fulfil when crewed. The relationships indicate that the system *can* create the flow and does not imply that it *must* create that flow. For example, the attack configuration Huey Helicopter has the *"isolated personnel position"*, *"isolated personnel status"* and *"situation information"* flows.

Matrix 6 – Determined System to Flow Relationships

	Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
Huey Helicopter (Rescue Configuration)		✓	✓	✓	✓	✓
Huey Helicopter (Attack Configuration)		✓	✓	✓		
Air Force Controllers - "Paddy Control"	✓		✓			
57th Medical Attachment Controllers - "Dustoff Operations"	✓					

With the system to flow relationships captured in Matrix 6 the system to system relationships as determined by the flows can be found from Figure 71, as captured in Matrix 7.

Matrix 7 – Indicated PSM System to System Relationships

	Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers
Huey Helicopter (Rescue Configuration)		✓		
Huey Helicopter (Attack Configuration)	✓			
Air Force Controllers - "Paddy Control"	✓			✓
57th Medical Attachment Controllers - "Dustoff Operations"	✓			

We now seek deeper insight into the system to system interconnections as informed by the transformation of the PIM flow to flow model (as determined in Matrix 5) into a PSM system to system model utilising the model transform determined in Matrix 6. The full transformation is shown in Matrix 8. The PIM is shown in the bottom left with the transform shown bottom right and the resultant PSM from the transformation shown top right.

Matrix 8 – PIM to PSM Transformation

						Huey Helicopter (Rescue Configuration)	✓	✓		
						Huey Helicopter (Attack Configuration)	✓			
						Air Force Controllers	✓	✓	✓	
						57th Medical Attachment Controllers	✓	✓	✓	
Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel		Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers	57th Medical Attachment Controllers
	✓	✓	✓	✓		Isolated personnel details			✓	✓
					✓	Situation information	✓	✓		
					✓	Isolated personnel position	✓	✓	✓	
					✓	Isolated personnel status	✓	✓		
	✓					Support information	✓			
						Rescued personnel	✓			

The resultant PSM system to system model, determined by the transformation shown in Matrix 8, is shown in Matrix 9. This profile may highlight missing relationships or help guide future development of the systems by identifying relationships useful for conducting particular missions. Note that the transformed PSM SoS relationships are more interconnected than the indicated relationships captured in Matrix 7. From Matrix 9 it can be observed that the Huey Helicopter (Attack Configuration) only has relationships with the Huey Helicopter (Rescue Configuration) and the Air Force Controllers and Huey Helicopter (Rescue Configuration) are highly coupled. There are no relationships back to the 57th Medical Attachment Controllers which implies a possible gap and indicated that its role may be redundant from the process and flow viewpoint. This is further supported as the only flow the 57th Medical Attachment Controllers is duplicated by the Air Force Controllers.

Matrix 9 – Transformed PSM System to System Relationships

	Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers
Huey Helicopter (Rescue Configuration)	✓	✓		
Huey Helicopter (Attack Configuration)	✓			
Air Force Controllers - "Paddy Control"	✓	✓	✓	
57th Medical Attachment Controllers - "Dustoff Operations"	✓	✓	✓	

Matrix 9 is the implied model as determined by the PIM flow to flow relationships transformed using the flow to system matrix. This only considers the full complement of systems used in the mission and does not consider subsets as SoS. To investigate the profiles of different SoS configurations on the PSM a transformation from PSM to PIM is required, as illustrated in Matrix 10.

Matrix 10 – PSM to PIM Transformation

Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers	57th Medical Attachment Controllers		Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
				Isolated personnel details		✓	✓	✓	✓	✓
				Situation information		✓	✓	✓	✓	✓
				Isolated personnel position		✓	✓	✓	✓	✓
				Isolated personnel status		✓	✓	✓	✓	✓
				Support information		✓	✓	✓		
				Rescued personnel		✓	✓	✓		
	✓			Huey Helicopter (Rescue Configuration)		✓	✓	✓	✓	✓
✓				Huey Helicopter (Attack Configuration)		✓	✓	✓		
✓				Air Force Controllers	✓		✓			
				57th Medical Attachment Controllers	✓					

The PSM to PIM transformation matrix shown in Matrix 10 allows different SoS configurations to be transformed to a flow to flow relationship matrix. The example shown is the minimal SoS configuration found whose PIM covers the model required for CSAR as shown in Matrix 5. As indicated previously the role of the 57th Medical Attachment Controllers seemed redundant, the profile generated by Matrix 10 shows this to be the case. The relationships can be determined by another PIM to PSM transformation as shown in Matrix 11.

Matrix 11 – PIM to PSM with Minimal SoS Transformation

						Huey Helicopter (Rescue Configuration)	✓	✓		
						Huey Helicopter (Attack Configuration)	✓			
						Air Force Controllers	✓	✓	✓	
						57th Medical Attachment Controllers				
Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel		Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers	57th Medical Attachment Controllers
	✓	✓	✓	✓		Isolated personnel details			✓	
					✓	Situation information	✓	✓		
					✓	Isolated personnel position	✓	✓	✓	
					✓	Isolated personnel status	✓	✓		
	✓					Support information	✓			
						Rescued personnel	✓			

The transformed PSM determined in Matrix 11 and shown in Matrix 12 is the resultant Mission Profile for this particular mission. This will be compared to the Commanders Intent Profile, which will be determined in the next subsection.

Matrix 12 – Mission Profile PSM

	Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers
Huey Helicopter (Rescue Configuration)	✓	✓		
Huey Helicopter (Attack Configuration)	✓			
Air Force Controllers - "Paddy Control"	✓	✓	✓	
57th Medical Attachment Controllers - "Dustoff Operations"				

7.2.2 Commanders Intent Profile

The narrative for the mission, previously given in subsection 4.3.1.1, is reconsidered in Table 9 and the flows (as identified from an *initiator* to a *recipient*) matched to the events.

Table 9 – Sequence of Events and Flow Mappings for Case Study One

No	Description	Flow	Initiator	Recipient
1	Paddy Control contacts Dustoff Operations with an emergency scramble mission - a Navy Seawolf helicopter had given a mayday call near Tra Vinh indicating that it had been hit by enemy fire and was going down.	Isolated personnel details	Paddy Control	Dustoff Operations
2	Captain David Freeman, a Dustoff pilot from the 57th Medical Detachment, and his crew scramble.	Isolated personnel details	57th Medical Attachment Controllers "Dustoff Operations"	Huey Helicopter Rescue Configuration
3	Paddy vectors Freeman to the coordinates where they had last seen the downed Seawolf on RADAR.	Isolated personnel position	Paddy Control	Huey Helicopter Rescue Configuration
4	Freeman takes approximately 15 minutes to get there.			
5	Freeman starts a search pattern which involves flying north to south with the crew visually searching for the crashed helicopter or a signal from their crew. At this stage Freeman does not know if anyone has survived or the tactical situation.			
6	Within a couple of minutes the crew chief spots and points out the wreckage of the helicopter. An initial assessment of the situation by	Isolated personnel position	Huey Helicopter Rescue Configuration	Huey Helicopter Rescue Configuration

No	Description	Flow	Initiator	Recipient
	Freeman and his crew is:			
7	The downed Seawolf is in an open area, easily visible and with plenty of room next to it for the Huey to land.	Situation information	Huey Helicopter Rescue Configuration	Huey Helicopter Rescue Configuration
8	The downed Seawolf looks in good physical condition with little external damage.			
9	From a tactical perspective the location is bad as the open area is difficult to defend with little cover.			
10	Freeman makes a low pass over the helicopter and sees four airmen. They are huddled next to the downed Seawolf and holding off a squad of Vietcong with their personal weapons. The downed Seawolf has a 50-calibre machine gun mounted on it but it was facing the wrong way and could not be swung around for defence.	Isolated personnel status	Huey Helicopter Rescue Configuration	Huey Helicopter Rescue Configuration
11	Freeman gets a call on UHF radio from another Seawolf helicopter which had responded to the emergency call. The pilot told them he was five minutes away and would give them gun cover whilst they made the pickup.	Situation information Situation information	Huey Helicopter Attack Configuration Huey Helicopter Rescue Configuration	Huey Helicopter Rescue Configuration Huey Helicopter Attack Configuration
12	Whilst waiting for the Seawolf to arrive the Dustoff crew decide on what appeared to be the best plan of action.			
13	Freeman approaches the crash site facing the enemy location and, using the downed helicopter as cover, land within 20-25 yards of where the downed crew were making their stand. As the downed crew ran towards the rescue helicopter the Seawolf arrived and fired on the enemy position with machine gun fire and rockets. This caused the Viet Cong to cease-fire and dig in, allowing the downed crew to scramble onto the Dustoff and Freeman exited the way that they came in without taking any hits.	Rescued personnel	Huey Helicopter Rescue Configuration	CSAR Operation

From Table 9 a modified diagram showing the flows between the CSAR systems can be drawn, as shown below in Figure 72. Note how the systems do not map one to one onto the processes as shown in Figure 70, but rather as distinct processes defined by their flows. The process used in this case study will seek to understand the relationship between this system of systems to the mission's flow diagram and in so doing identify any gaps or shortfalls in the capability of that SoS.

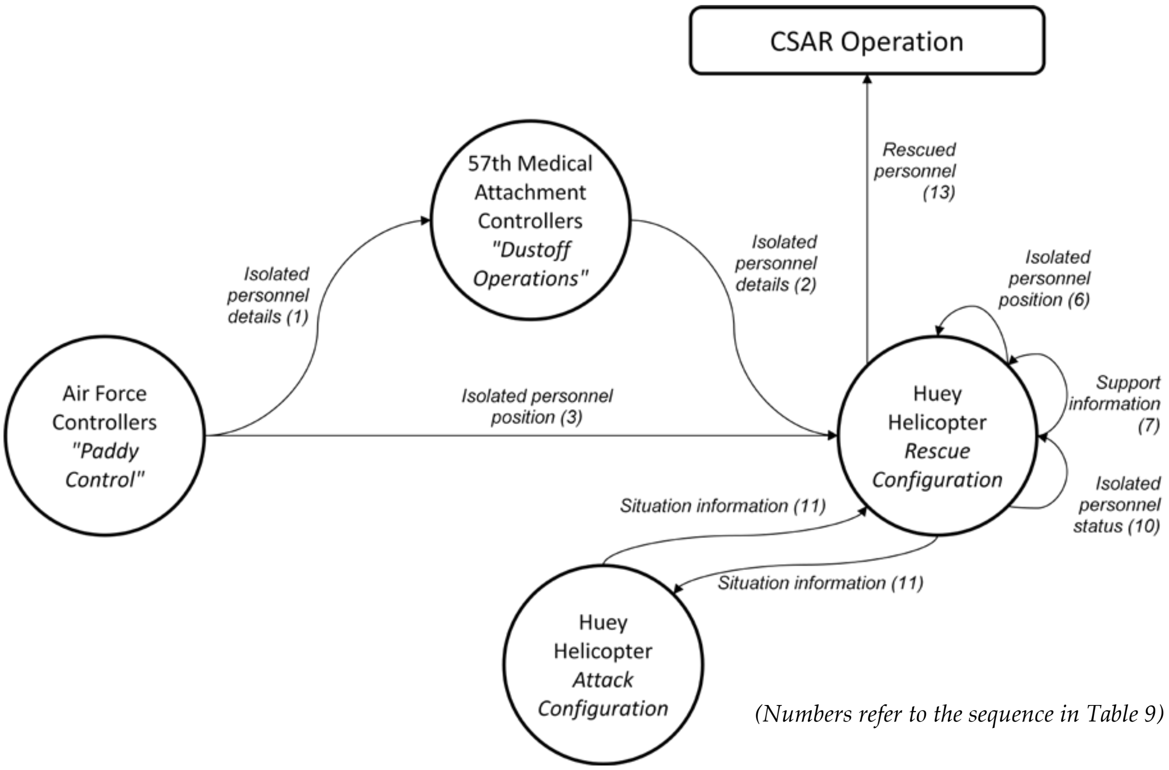


Figure 72 – Case Study One Flow Diagram

With the commanders intent captured in Figure 72 the relationships between the available systems and the flows can be captured as shown in Matrix 13. These relationships are captured by identifying the systems which originate the flows (the recipients will be passively identified through the system-to-system relationships later on in this process).

Matrix 13 – RPDM System to Flows Relationships

	Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
Huey Helicopter (Rescue Configuration)		✓	✓	✓	✓	✓
Huey Helicopter (Attack Configuration)		✓				
Air Force Controllers - "Paddy Control"	✓					
57th Medical Attachment Controllers - "Dustoff Operations"	✓		✓			

For the decision makers viewpoint the Figure 72 diagram can be used to determine the relationships between the systems which are captured in Matrix 14. These are the implied relationships which would be validated to ensure the requirements of the mission, as expressed in the CIM and PIM, are realised. The validation of these relationships is beyond the scope of the technical information available as ultimately it would require a review by subject matter experts to review who are unavailable for this research. However, the generated profile from the historical narrative will suffice for comparison with the mission viewpoint PSM and the purposes of this case study.

Matrix 14 – Commanders Intent Profile PSM

	Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers
Huey Helicopter (Rescue Configuration)	✓	✓		
Huey Helicopter (Attack Configuration)	✓			
Air Force Controllers - "Paddy Control"	✓			✓
57th Medical Attachment Controllers - "Dustoff Operations"	✓			

With the Commanders Intent Profile established this can be compared to the Mission Profile, which is the focus of the next subsection.

7.2.3 Profiles Comparison

The PSM profiles from the mission viewpoint and the commander viewpoint are compared in Matrix 15.

Matrix 15 – PSM Profile Comparison

Mission Viewpoint					Commanders Viewpoint			
Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers - "Dustoff Operations"		Huey Helicopter (Rescue Configuration)	Huey Helicopter (Attack Configuration)	Air Force Controllers - "Paddy Control"	57th Medical Attachment Controllers - "Dustoff Operations"
✓	✓			Huey Helicopter (Rescue Configuration)	✓	✓		
✓				Huey Helicopter (Attack Configuration)	✓			
✓	✓	✓		Air Force Controllers - "Paddy Control"	✓			✓
				57th Medical Attachment Controllers - "Dustoff Operations"	✓			

The most notable feature comparing the mission viewpoint and the commanders viewpoint is that the mission viewpoint does not utilise the 57th Medical Attachment Controllers at all. Looking back at the commanders viewpoint of the systems illustrated in Figure 72 it can be seen that the role of the 57th Medical Controllers is redundant in the context of this particular CSAR Operation. It can be seen from Matrix 15 that the commanders viewpoint of the SoS to be used is based around the Huey Rescue Helicopter. This is shown by the relationships from it to all other systems and vice versa. In contrast the mission viewpoint suggests a decentralised and more interconnected SoS with more relationships between the reduced set of systems.

In terms of relationship differences between the two viewpoints the mission viewpoint removes all relationships on the 57th Medical Attachment Controllers, as discussed previously. A new relationship from the transformed mission viewpoint PIM to PSM is from the Air Force Controllers to the Huey helicopter (attack configuration). In Figure 71, which shows all of the possible flows between systems, there are no flows between the Air Force Controllers and the Huey helicopter (attack configuration). However, it can be seen that the flow *isolated personnel position* is routed from the Air Force Controller via the Huey helicopter rescue configuration to the Huey helicopter attack configuration. An opportunity has been identified to direct this flow from the originator (Air Force Controllers) direct to the recipients without an intermediary. Hence the commanders viewpoint only allows communication between the Huey helicopter rescue configuration and the Huey helicopter attack configuration. This means that all information from the Air Force Commanders must be relayed by the Huey helicopter rescue configuration to the Huey helicopter attack configuration. The mission viewpoint has identified an opportunity to communicate directly from the Air Force Controllers to the Huey helicopter attack configuration and in so doing relieving the Huey helicopter rescue configuration from having to relay that information. This was identified through the transformed mission viewpoint PIM to PSM. Note that this does not impinge on the agile force structure of the commanders intent, it simply reinforces it by unburdening the rescue configuration Huey helicopter from relaying messages from the Air Force Controllers to the attack configuration Huey helicopter.

The second new relationship is a reflexive relationship on the Air Force Controllers. This new relationship provides the insight that the systems role may have increased beyond a peripheral system into a more central role requiring self reference to fulfil more than one process. This indication points to a specific relationship that may require further analysis to understand the implications of that relationship.

7.2.4 Case Study One Revisited Conclusions

The reanalysis of this case study using the updated DSS has shown that the SoS used to conduct the mission was suitable in terms of flows and processes. The analysis has revealed that the involvement of the 57th Medical Attachment Controllers was redundant and higher levels of capability may have been realised by decentralising the SoS away from the rescue configuration Huey helicopter to a smaller, more interoperable SoS. This disagrees with the finding from the first consideration of this case study in 4.3.1 that only the combination of all four available systems could successfully complete the mission. Revisiting the results generated by the first DSS tool show that the Dustoff Operations fulfilled the *select appropriate resources* and *prepare resources* functions within the *Planning & Adapting* functional grouping of the original generic functional model (GFM). Considering the boundary of the problem space the DSS considers, as identified in the developed flow diagram for CSAR Operations previously shown in Figure 63, these functions are not within the remit of the SoS to conduct the mission. The precise profiling of the SoS boundary through this model clearly delineates between the conduct of the mission and the support infrastructure pre and post mission. The model driven approach for a DSS has revealed the underlying processes required for conduct of the mission and provided scope for simplification and greater networking within the reduced SoS. The results of this case study has falsified the parametric approach and identified improvements that could be made on the commanders intent.

7.3 Case Study Two Revisited – Rescue Of Lt Devon Jones

This subsection revisits the case study previously conducted in subsection 4.3.2, Case Study Two – Rescue Of Lt Devon Jones. The narrative for this case study was previously given in subsection 4.3.2.1. This case study was repeated using the methodology shown previously in Chapter 6 and sought to understand the implications of the various SoS configurations against mission requirements.

The four systems profiled for this case study are:

- MH-53J 'Pave Low' helicopter
- F-15C 'Eagle' tactical fighter aircraft
- 'Sandy' A10A close support aircraft
- AWACS (Airborne Warning And Control System)

As with the previous case study the developed flow diagram for CSAR operations developed in section 6.4 will be used as a template for conducting CSAR operations and modified diagrams will be developed for this case study based on the roles and interactions of the systems in this case study. The developed DSS tool required two input profiles, the *Standard Flow Profile* and the *Commanders Profile* which are the focus of the next subsections.

7.3.1 Mission Profile

As for the previous case study the Standard Flow Profile is used to capture all the possible connections between the available systems by modelling *flows*. While the flows were explicitly shown as inter systems flows in the previous case study a different graphical approach will be used for this case study due to the higher level of flow interconnections. Rather than showing the flows as dedicated lines as for the first case study only the flows that each system can produce have been identified. This is a more efficient approach as the relationships between individual systems do not need to be captured, this will be determined by the DSS. The flows have been modelled by capturing flow originators as illustrated in Figure 73.

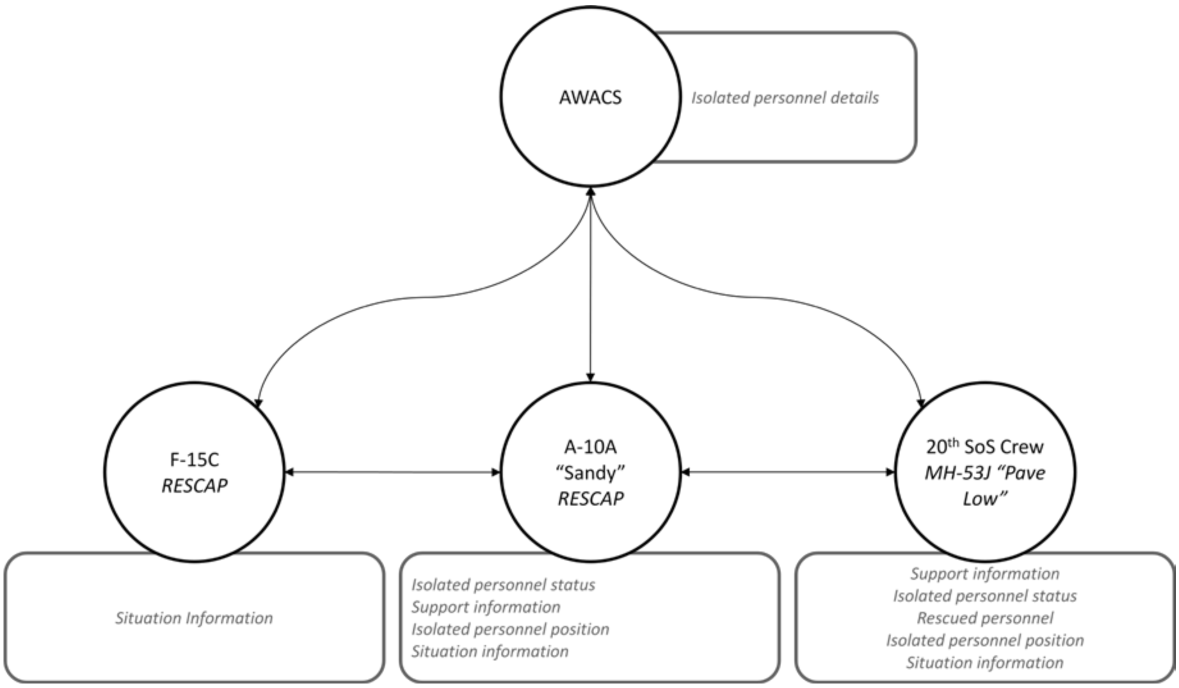


Figure 73 – System Flow Profile

From Figure 73 the identified system to flow relationships can be expressed as a matrix, as shown in Matrix 16.

Matrix 16 – Mission System to Flow Relationships

	Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
AWACS	✓					
F-15C RESCAP		✓				
20th SoS Crew MH-53J "Pave Low"		✓	✓	✓	✓	✓
A-10A "Sandy" RESCAP		✓	✓	✓	✓	

From Figure 73 the identified system to flow relationships can be expressed as a matrix, as shown in Matrix 16. Note that the F-15C RESCAP to 20th SoS Crew MH-53J Pave Low relationship is implied in Figure 73 but not explicitly shown for layout reasons. Essentially every system can have a relationship with any other system. This is in contrast to the more stovepiped systems considered in Case Study One.

Matrix 17 – Mission Viewpoint System to System Relationships

	AWACS	F-15C RESCAP	20th SoS Crew MH-53J “Pave Low”	A-10A “Sandy” RESCAP
AWACS		✓	✓	✓
F-15C RESCAP	✓		✓	✓
20th SoS Crew MH-53J “Pave Low”	✓	✓		✓
A-10A “Sandy” RESCAP	✓	✓	✓	

The relationships shown in Matrix 17 are all of the possible relationships, but which ones are required by the CSAR mission? To determine this requires the established PIM for CSAR missions, captured previously in Matrix 5, to be transformed using the systems to flow relationships established in Matrix 16. The overall transformation is shown in Matrix 18.

Matrix 18 – PIM to PSM Transformation

						AWACS		✓	✓	✓
						F-15C RESCAP			✓	
						20th SoS Crew MH-53J "Pave Low"		✓	✓	✓
						A-10A "Sandy" RESCAP		✓	✓	✓
Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel		AWACS	F-15C RESCAP	20th SoS Crew MH-53J "Pave Low"	A-10A "Sandy" RESCAP
	✓	✓	✓	✓		Isolated personnel details	✓			
					✓	Situation information		✓	✓	✓
					✓	Isolated personnel position			✓	✓
					✓	Isolated personnel status			✓	✓
	✓					Support information			✓	✓
						Rescued personnel			✓	

The PSM determined by the transform of the CSAR PIM is a subset of the possible system connections captured previously in Matrix 17. Note how there are no relationships back on the AWACS. As the AWACS is the only system that produces the *isolated personnel details* flow this is acceptable. Thus the PSM determined in Matrix 18 is the Mission Profile PSM, as shown separately in Matrix 19.

Matrix 19 – Mission Profile PSM

	AWACS	F-15C RESCAP	20th SoS Crew MH-53J "Pave Low"	A-10A "Sandy" RESCAP
AWACS		✓	✓	✓
F-15C RESCAP			✓	
20th SoS Crew MH-53J "Pave Low"		✓	✓	✓
A-10A "Sandy" RESCAP		✓	✓	✓

This completes the necessary standard flow profile for input into the DSS tool. As an aside, note that the only consideration here is what flows the systems are capable of producing. These would, in practise, be evaluated by subject matter experts and, more importantly, maintained by them as the systems are upgraded and improved. Much like the first approach this one also supports a *catalogue* of systems which can be iteratively maintained and provides a ready source of system profiled to speed up the approach. This second profile required for input into the DSS is the commanders intent profile which is the focus of the next subsection.

7.3.2 Commanders Intent Profile

The narrative for the mission, previously given in subsection 4.3.2.1, is reconsidered in Table 10 and the flows (as identified from an *initiator* to a *recipient*) matched to the events.

Table 10 – Sequence of Events and Flow Mappings for Case Study Two

No	Description	Flow	Initiator	Recipient
1	AWACS receives mayday call from Slade as he parachutes down after ejecting from his aircraft.			
2	Jones and Slade land some distance away from each other			
3	Jones digs in and camouflages himself			
4	MH-53J 'Pave Low' helicopter scrambled to crash site	Isolated personnel details	AWACS	MH-53J 'Pave Low' F-15C 'Eagle' 'Sandy' A10A
5	Iraqi MiG-23 fighters appear, engaged by two USAF F-15Cs on RESCAP (Combat Air Patrol), MiG-23's retreat	Situation information	F-15C 'Eagle'	MH-53J 'Pave Low'
6	At approximately 10:30 Jones hides from what he thinks are enemy fighters, actually the two F-15Cs on RESCAP			
7	At approximately 10:30 Slade is discovered and detained by Iraqi forces			
8	MH-53J 'Pave Low' helicopter returns to Arar to refuel, returning immediately to continue search			
9	'Sandy' A-10A makes contact with Jones via radio on SAR frequency	Support information	'Sandy' A10A	'Sandy' A10A
10	Jones guides the 'Sandy' A-10A to him, 'Sandy' A-10A locates him exactly using its INS	Isolated personnel position Isolated personnel position	'Sandy' A10A 'Sandy' A10A	'Sandy' A10A MH-53J 'Pave Low'
11	Iraqi MiG-23s move into area, engaged by two USAF F-15Cs on RESCAP (Combat Air Patrol), MiG-23's retreat	Situation information	F-15C 'Eagle'	MH-53J 'Pave Low'
12	'Sandy' A-10's sanitise the area, MH-53J 'Pave Low', returned from refuelling, waits on the ground	Situation information	'Sandy' A10A	MH-53J 'Pave Low'
13	MH-53J 'Pave Low' moves in to rescue Jones, spots an Iraqi truck en route to Jones' position	Situation information	MH-53J 'Pave Low'	'Sandy' A10A
14	'Sandy' A-10A engages Iraqi truck	Situation information	'Sandy' A10A	MH-53J 'Pave Low'
15	MH-53J 'Pave Low' lands near destroyed Iraqi truck, Jones breaks cover and boards the helicopter, helicopter returns to base	Rescued personnel Isolated personnel status	MH-53J 'Pave Low' MH-53J 'Pave Low'	CSAR Operation MH-53J 'Pave Low'
16	'Sandy A-10A's stay on station to assure the MH-53J's safety	Situation information	'Sandy' A10A	MH-53J 'Pave Low'

The mapping of flows to systems established through the sequence of events shown in Table 10 are shown diagrammatically in Figure 74.

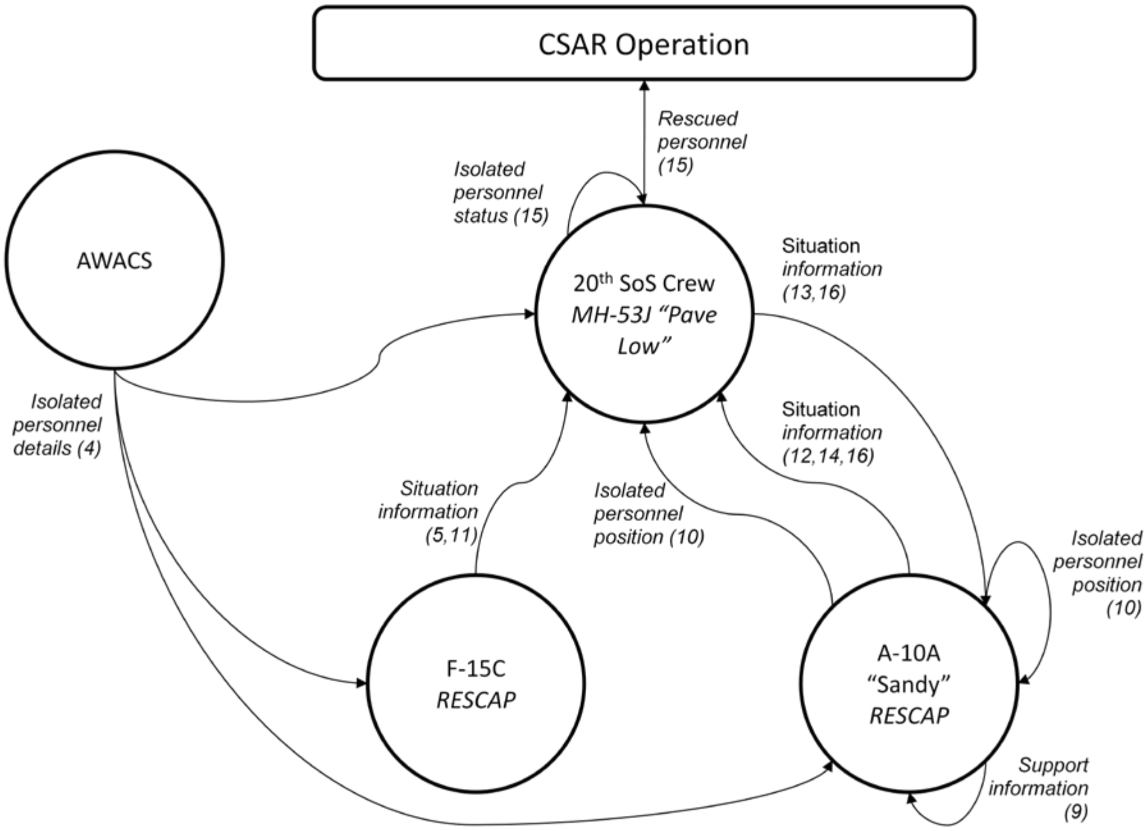


Figure 74 – Case Study Two Flow Diagram

With the commanders intent captured in Figure 74 the relationships between the available systems and the flows can be captured as shown in Matrix 20. These relationships are captured by identifying the systems which originate the flows (the recipients will be passively identified through the system to system relationships later on in this process).

Matrix 20 – RPDM System to Flow Relationships

	Isolated personnel details	Situation information	Isolated personnel position	Isolated personnel status	Support information	Rescued personnel
AWACS	✓					
F-15C RESCAP		✓				
20th SoS Crew MH-53J "Pave Low"		✓		✓		✓
A-10A "Sandy" RESCAP		✓	✓		✓	

Finally the system-to-system relationships are determined by considering Figure 74 in terms of the relationships between the systems which are captured in Matrix 21.

Matrix 21 – RPDM System to System Relationships

	AWACS	F-15C RESCAP	20th SoS Crew MH-53J “Pave Low”	A-10A “Sandy” RESCAP
AWACS		✓	✓	✓
F-15C RESCAP			✓	
20th SoS Crew MH-53J “Pave Low”			✓	✓
A-10A “Sandy” RESCAP			✓	✓

With the Mission Profile and Commanders Intent Profile established the profiles can be compared, which is presented in the next section.

7.3.3 Profiles Comparison

The PSM profiles are shown for comparison in Matrix 22. It can be seen that the Commanders Viewpoint Profile is identical to the Mission Profile and hence the appropriate relationships between systems have been utilised for the mission.

Matrix 22 – PSM Profile Comparison

Mission Viewpoint					*Commanders Viewpoint*			
AWACS	F-15C RESCAP	20th SoS Crew MH-53J “Pave Low”	A-10A “Sandy” RESCAP		AWACS	F-15C RESCAP	20th SoS Crew MH-53J “Pave Low”	A-10A “Sandy” RESCAP
	✓	✓	✓	AWACS		✓	✓	✓
		✓		F-15C RESCAP			✓	
	✓	✓	✓	20th SoS Crew MH-53J “Pave Low”			✓	✓
	✓	✓	✓	A-10A “Sandy” RESCAP			✓	✓

In practice this comparison would be used as validation of the commanders intent and give support to the decision maker that the SoS formed is as complete as can be and meets the missions requirements.

7.3.4 Case Study Two Revisited Conclusions

The reanalysis of case study two using the updated DSS has shown that the SoS used to conduct the mission was ideally suited in terms of flows and processes. The analysis has revealed that no subset of the systems involved could have been used. This disagrees with the (incorrect) findings of the original DSS which indicated that the F-15C RESCAP was redundant. In practice the analysis produced by the model driven approach for a DSS shown in this subsection would be taken as support for the Commanders Intent.

7.4 Evaluation of Model Driven Approach

The two case studies have been repeated with the updated model driven approach as implemented in a second prototype software tool.

There were three findings from the model driven approach for case study one, the Seawolf CSAR in Vietnam. The first finding was that one of the component systems, the 57th Medical Attachment Controllers, was unnecessary. The parametric approach had not identified this redundant system and had incorrectly indicated that all of the systems were required to conduct the mission. This result alone falsifies the parametric approach. The second finding was a new relationship from the Air Force Controllers to the Huey helicopter (attack configuration). An opportunity had been identified to avoid utilising a pivotal system, the Huey helicopter (rescue configuration) as a message relay. The mission viewpoint had identified an opportunity to communicate directly from the Air Force Controllers to the Huey helicopter attack configuration and in so doing relieving the Huey helicopter rescue configuration from having to relay that information. This was identified through the transformed mission viewpoint PIM to PSM. Note that this does not impinge on the agile force structure of the commanders intent, it simply reinforces it by unburdening the rescue configuration Huey helicopter from relaying messages from the Air Force Controllers to the attack configuration Huey helicopter. The third new relationship identified was a reflexive relationship on the Air Force Controllers. This provided an indication that the systems role may have increased beyond a peripheral system into a more central role requiring self reference to fulfil more than one process. This was in contrast to the parametric approach which had incorrectly indicated that all of the systems were needed. The results of this case study have falsified the parametric approach and identified improvements that could be made on the commanders intent.

The model driven approach for Case Study Two found that all of the component systems were required to conduct the mission which was the same as the commanders intent. The parametric approach found that the not all of the systems were required, which was incorrect and falsified this approach. The model driven approach identified that all systems were required and verified the relationships between them from the functional perspective. However, the model driven approach only utilised the functional set of characteristics, no consideration was given to the non-functional and physical environment characteristics as the original parametric approach did. This is discussed further below in the feasibility of this approach.

At the start of this second set of case studies three main aims were set to be achieved by developing the second prototype software tool:

1. **Feasibility** - demonstrate that the developed DSS architecture could be implemented in a software tool.
2. **Validation** - perform a set of historical case studies utilising the developed prototype software tool.
3. **Guide** - guide the further development of the architecture of an approach for a DSS.

The feasibility of the model driven approach has been demonstrated through the case studies which have successfully applied the approach. The findings from the case studies were more informative and insightful than those from the original parametric approach. This is due to the model driven approach which has separated the systems and functionality. These were intertwined in the previous parametric approach and made forming and assessing alternate SoS difficult. The model driven approach has separated the systems and functionality through the use of Model Driven Architecture (MDA) which establishes a platform independent model that captures the functionality required in such a way that he ultimate implementation of that functionality can be migrated across technology. Through the success of the second set of case studies the model driven approach has not be falsified for functional characterisation of missions and SoS. However, unlike the original parametric approach no consideration of the non-functional and physical environment characteristics was made with the model driven approach. Hence while the model driven approach has not been falsified by the functional profile (which the parametric approach was falsified against) it may yet be falsified against the non functional and physical environment characteristics.

A number of ideas were generated from the development of the second implementation using the architecture of a model driven approach for a DSS. These will be discussed in more detail in the Future Work section (8.4) of the next Chapter 8.

The utility and value of this approach in supporting the commanders decision through the RPD model has been shown through the case studies findings. The DSS is not intended to be prescriptive, rather its role is more of a guard function to protect against oversights and errors.

In line with the research methodology outlined previously in section 1.5 the epistemology used in this research is critical rationalism. In line with this the developed architecture of an approach for a DSS cannot be *proved*, but rather its use can be demonstrated and through that demonstration it can be falsified. The first developed prototype tool and the parametric approach it was based on was falsified. This tool, within the limits of the two case studies conducted, has not been.

7.5 Towards Generic Mission Models

This section explores an extension of the Model Driven Approach discovered through analysis of the case studies. Consider the Mission Viewpoint system to flow profiles shown previously for case study one in Matrix 6 and for case study two in Matrix 16 and shown as a side by side comparison in Matrix 23.

Matrix 23 – Mission Viewpoint System to Flow Profile Comparison

Case Study One					*Case Study Two*			
57th Medical Attachment Controllers - "Dustoff Operations"	Air Force Controllers - "Paddy Control"	Huey Helicopter (Attack Configuration)	Huey Helicopter (Rescue Configuration)		AWACS	F-15C RESCAP	A-10A "Sandy" RESCAP	20th SoS Crew MH-53J "Pave Low"
✓	✓			Isolated personnel details	✓			
		✓	✓	Situation information		✓	✓	✓
	✓	✓	✓	Isolated personnel position			✓	✓
		✓	✓	Isolated personnel status			✓	✓
			✓	Support information			✓	✓
			✓	Rescued Personnel				✓

Note in Matrix 23 that while the composition of systems is different for each case study there appears to be a pattern with respect to their relationships to the flows. It can be seen for example that both the 57th Medical Attachment Controllers and AWACS only have a single relationship to the *isolated personnel details* flow. Similarly it can be seen that the Huey Helicopter (Rescue

Configuration) and 20th SoS Crew MH-53J "Pave Low" have the same relationship flows. It can also be seen that the Huey Helicopter (Attack Configuration) and both the F-15C RESCAP and A-10A "Sandy" RESCAP have a level of commonality between them.

This implies that there may be a level of commonality between the two different case study instantiations of the CSAR mission in terms of system to flow relationships. This indicates that the findings of the GFM developed in the parametric approach were correct in so far that there exists a level of commonality at the higher levels between all CSAR missions. It should be stressed that this is an unproven hypothesis and further testing will be required to validate this. If shown to be true then the original GFM may be reusable for future work.

The utility of this finding, once confirmed, is that verification of a SoS is only needed once against the functional perspective through the identification and allocation of roles (as defined by flow relationships). The clear identification of the role of a system by its relationships to flows gives the commander a view on what they can do with available systems. This will also enable automated profiling of systems for identified mission roles. An extension of this research thread would consider the allocation of the previously developed primary and secondary attributes and factors onto the commander's intent to differentiate between functionally identical SoS.

Within the boundary of CSAR missions it appears that a generic functional viewpoint of the mission has been generated. My belief is that I have tamed of some elements of the CSAR wicked problem by representing some elements of CSAR in an objective manner.

7.6 Chapter Summary

This chapter has revisited the case studies conducted earlier in this thesis using the new model driven approach developed in Chapter 5 and implemented in a software tool in Chapter 6. The two case studies originally conducted in Chapter 4 with the original parametric approach were revisited using a model driven approach. For each of the case studies a new Mission Profile and Commanders Intent Profile were generated. These profiles were compared and a set of conclusions made on the analysis. These conclusions revealed deeper insights into the structure of the SoS for the mission than the parametric approach could identify. This analysis approach offered a significant improvement on the previous set of case studies conducted with the original DSS tool. Finally, in light of the revisited case studies an overall evaluation of the tool and the model driven approach was given. The tool was judged to have been a success, validating the model driven approach and the demonstration through the case studies did not result in a falsification of the approach.

Chapter 8
Summary and Conclusion

Outline of Chapter

Section 8.1 of this Chapter summarises the research conducted in this thesis and discusses the development of the architecture of the approach for a decision support system that was used. The parametric and model driven approaches developed in this research are reviewed in subsections 8.1.2 and 8.1.3 respectively. The approaches are demonstrated through case studies in subsection 8.1.4. The research is reviewed against the objectives in section 8.2 and against the research problem in section 8.3. The research contributions of this research are discussed in section 8.4 and further work opportunities identified in section 8.5. Finally the research and this thesis is drawn to an end with a conclusion in section 8.6.

Overview of the Contributions of this Research

*This **Research** established a **Problem** and **Objectives** to develop an **Architecture** of a **Model Driven Approach** for a **Decision Support System**. The approach matches **System of Systems** to **Military Missions**. This Research references a **Literature Review**.*

*The decision support system supports the **Decision Maker** through the Recognition Primed Decision model. The **Application** for this research is the **Combat Search and Rescue** military mission.*

*The decision support system supports a **Decision Maker** by characterising military missions and system of systems with a set of **Common Characteristics** which informs the decision maker.*

*In the model driven approach the system of systems is captured in a **Platform Specific Model**, the common characteristics are captured in a **Platform Independent Model** and the combat search and rescue mission is captured in a **Computational Independent Model**. These models can be related by **Model Transformation**. The architecture has been implemented in a **Software Tool**.*

*The software tool was used to conduct a set of literature based **Case Studies** of the application of this research (combat search and rescue) and the results evaluated.*

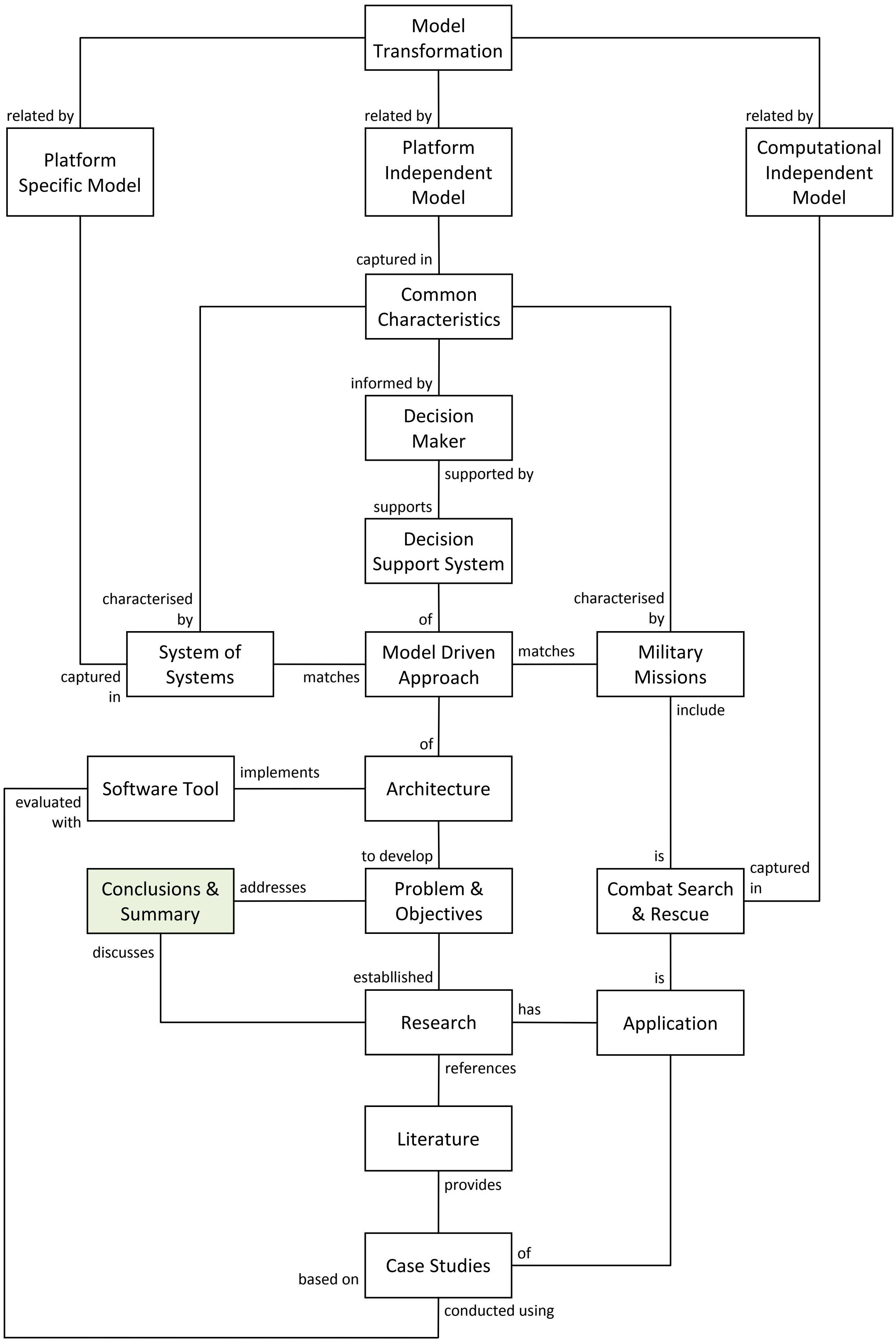


Figure 75 – Logical Model of Research Contributions of Chapter Eight

8.1 Summary of Research

This chapter evaluates the research conducted and reported in this thesis. The two areas of focus are firstly on the developed decision support system and secondly on the case studies conducted using the decision support system.

This thesis started with a discussion of the nature of systems problems and the need for an open viewpoint to explain a system by viewing it as part of a larger whole and explaining its role in terms of that larger whole. This was linked to General Systems Theory which sought to provide a common language to discuss systemic problems by combining modelling and communication; modelling to provide precision while communication provided comprehensibility. An overview of the definition of *system* and *system of systems* was covered and led to a discussion of *wicked problems*. A bridge was formed between the open system viewpoint of the architect and the closed system viewpoint of the engineer, which provided a common language that General Systems Theory sought. This allowed the wicked planning problem of the whole to be addressed using an open system viewpoint while still maintaining a bridge to the closed system viewpoint required to solve individual problems. Overall this approach was an application of a "synthetic mode of thought" to systems problems (Ackoff 1974, 3) and hence was a *systems approach*. A practical application for the research was discussed and a need identified for this approach within the military domain. From this the research problem was stated:

Is there a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission?

Statement of Research Problem

A number of objectives were set to help answer this research problem as outlined in detail in section 1.6. These objectives are listed below, for the description of each see section 1.6:

Objective 1: Review General Decision Making
Objective 2: Review Military Decision Making
Objective 3: Identify a Decision Making Model to Support
Objective 4: Identify a Practical Application for this Research
Objective 5: Develop an Architecture of an Approach for a Decision Support System
Objective 6: Implement the Developed Approach
Objective 7: Conduct a Set of Case Studies with the Implemented Approach

The following subsections 8.1.x summarise the research and do not speak to the objectives directly. The objectives will be addressed in section 8.2 where it will be shown how the research achieves the objectives. Subsection 8.1.1 summarises the decision making research which covers objectives 1 to 4. Objectives 5 and 6 were to develop and implement an architecture of an approach for a decision support system. These are covered by the two distinct approaches developed in this research, a parametric approach which is discussed in subsection 8.1.2 and a model driven approach discussed in subsection 8.1.3. Subsection 8.1.4 summarises the case studies which covers objective 7.

8.1.1 Decision Making Summary

Decision making under uncertainty was considered in section 2.2 and a descriptive model of military decision makers was sought for which a decision support system could be developed. Two main decision making theories from economics were the *Expected Utility Theory* which states that people maximise utility as opposed to reward and *Prospect Theory* which characterises the general behaviour of decision makers with the terms Loss Aversion and Diminishing Sensitivity. Both of these theories indicated that humans are innately irrational decision makers liable to subject to a number of decision making fallacies. Note that these do not specify how decision makers actually make decisions, rather it characterised the likely features of the decision made. A decision support system would be useful to help negate these decision biases.

How decisions are currently made within the military was reviewed in section 2.3, starting with the role of information was considered and it's increasingly important role in redefining how superiority over an enemy can be defined in terms of decision making. This found that military decision making is typically time constrained and highly stressful which limit human decision making and can negate any informational advantage. A decision support system which could alleviate these effects and maintain any informational advantage would be useful.

The history of decision making in military operations was traced in subsection 2.3.1 and the Military Decision Making Process was discussed in subsection 2.3.2 and the Observe, Orient, Decide and Act Loop was discussed in subsection 2.3.3. Emerging tenets such as Network Centric Warfare (NCW) and its impact on situational awareness and command and control were presented in subsection 2.3.4. The proliferation of decision makers led to alternatives to the classical decision making theories as discussed in section 2.4. These alternatives mostly emanated from descriptive studies of decision making in naturalistic environments. Naturalistic Decision Making is the description of proficient decision making strategies that emphasise recognitional as opposed to analytical processes. A leading theory of Naturalistic Decision Making is Klein's Recognition Primed Decision (RPD) strategy which was selected as the most representative model of how military decision makers actually make decisions and the decision making model that this research would support.

The Combat Search And Rescue (CSAR) mission was selected as the particular military mission that this decision support system would support in section 2.5. This mission was chosen as it is a time sensitive reactive mission requiring an SoS to be rapidly assembled to conduct the mission. With a practical application for this research established the next chapter considers the approaches developed for a decision support system.

8.1.2 Parametric Approach Summary

The first parametric approach sought to understand how military missions and system of systems could be understood in similar and comparable terms by using common sets of characteristics. Three different sets of common characteristics were identified: functional, non-functional and physical environment. Sets of characteristics for each were identified using various techniques through group work. The functional characteristics were captured through the development of a Generic Functional Model (GFM) using Functional Flow Diagrams (FFD) and Viewpoint Analysis (VPA), as discussed in section 3.2. The development of the GFM aimed to capture the high level functionality required by the majority of CSAR missions and was verified through available doctrines as discussed in section 3.2.6. The non-functional characteristics are often referred to as the *ilities*, but they are poorly defined, difficult to measures and specifically understood for particular systems and hence they were not usable in their current form. A hierarchical set of terms and definitions were developed to overcome these limitations: Attributes, Secondary Attributes and Factors as presented in section 3.3. The physical environment characteristics were captured in section 3.4 in a set which aimed to capture the pertinent environmental features of the mission that could be measured for available systems as well. With the three common sets of characteristics developed the overall approach to allow missions and SoS profiles to be matched was developed in section 3.5. In the context of this research a profile was specifically defined in section 3.1 as *a representation of the structured set of common characteristics of a mission or a system of systems*. The parametric approach was implemented in a prototype software tool in section Chapter 4. This tool implemented the mechanisms to create comparable mission and system profiles, as detailed in section 4.4.1 and the method for combining systems into SoS was defined in section 4.4.2.

A set of case studies were conducted using the developed parametric approach as presented in section 4.3. The findings from the case studies were incorrect and not useful to a decision maker, as discussed in section 4.4. The principle issue with the approach was the intertwining of systems and functionality. This made it difficult to combine system profiles into SoS profiles. An identified way forward was to shift from a parametric approach to a model driven approach, the development of which is the focus of the next subsection.

8.1.3 Model Driven Approach Summary

A number of key concepts were discussed in the development of a model driven approach, principally these were Transformational Grammar, Logical Modelling, Model Driven Architecture and model transformations. Transformational Grammar was presented in section 5.2 as a specific approach of generative grammar in theoretical linguistics. Transformational Grammar identifies two levels of representation of natural language sentences – the surface structure and the deep structure. An approach to elicit the deep structure from the surface structure of natural language in a model usable by an engineering process was required as the tree structure used by linguists was not appropriate. The Logical Modelling approach, which creates both surface structure and deep structure models of natural language using the Unified Modelling Language (UML), was identified in section 5.3 as appropriate for this research. In the software engineering domain the Model Driven Architecture has been identified as an approach which implements the model transformation concepts of Transformational Grammar for engineering applications. As discussed in section 5.4 principles from MDA can be applied to the systems engineering domain that this research addresses.

Model Driven Architecture is a realisation of the principles of structured analysis and design and achieves separation of the problem specification from the solution through the logical separation of the Computational Independent Model (CIM)/ Platform Independent Model (PIM) and Platform Specific Model (PSM) models. A lower Platform Model (PM) as specified by the MDA has been considered but is beyond the scope of this research and the approach has been bounded to the CIM/PIM/PSM models. MDA prescribes model transforms to move between these models, as described in section 5.4.1, and a methodology was presented in section 5.4.2 to achieve this whilst maintaining conceptual integrity through the use of mathematical matrix transforms. This has been captured in a developed software tool. Whilst this process provides a solution in the form of the PSM (or even PM) it does not provide a repeatable input methodology to the CIM. A method to achieve this was presented in section 5.5 to allow Logical Models to be converted into flow diagrams which in turn can be converted into a CIM and PIM. A coherent and complete approach pulling together all of the concepts discussed was presented in section 5.5 which formed a model driven approach. This approach was implemented in a software tool as described in Chapter 6.

With a model driven approach developed a set of case studies were conducted using the implemented tool which is the focus of the next subsection.

8.1.4 Model Driven Approach Case Study Summary

A set of case studies were conducted with the developed model driven approach, as described in Chapter 7. The evaluation of the case studies was shown in Chapter 8.

The model driven approach findings of case study one were that not all of the systems were needed and that unused relationships were identified between and on systems within this reduced SoS. This was in contrast to the parametric approach which had incorrectly indicated that all of the systems were needed. The results of this case study falsified the parametric approach and identified improvements that could be made on the commanders intent.

The model driven approach for Case Study Two found that all of the component systems were required to conduct the mission which was the same as the commanders intent. The parametric approach found that the not all of the systems were required, which was incorrect and again falsified the parametric approach. The model driven approach identified that all systems were required and verified the relationships between them.

The concise nature of the PSM models used for comparison made this set of case studies easier to interpret compared to the verbose output of the parametric approach. The main issue with the parametric approach, the intertwining of functionality with systems was addressed through the use of Model Driven Architecture (MDA) which establishes a platform independent model that captures the functionality required in such a way that the ultimate implementation of that functionality can be migrated across technology. Hence the functionality is separated from implementation.

In line with the research methodology outlined previously in section 1.5 the epistemology used in this research is critical rationalism. In line with this the developed architecture of an approach for a DSS cannot be *proved*, but rather its use can be demonstrated and through that demonstration it can be falsified. The first developed prototype tool and the parametric approach it was based on was falsified. This tool, within the limits of the two case studies conducted, has not been.

8.1.5 Model Driven Approach Limitations

As implemented in this research the model driven approach developed has a number of limitations, both in the context of this research and in general. First and foremost the model driven approach only considered the functional set of characteristics and not the non-functional and physical environment characteristics as established in Chapter 3. The reasoning behind this was that the functional characteristics had been the undoing of the original parametric approach and, due to time constraints, were the only ones considered for implementation in the model driven approach. To be used in the model driven approach both the non-functional and physical environment characteristics would need to be logically modelled to provide a logical pathway from the high level characterisation that the mission can be described by down to the lower system level where these characteristics can be met by individual systems. There would be significant reuse of the original characterisation work done in Chapter 3, it would be restructured and strengthened through logical modelling. The original functional characterisation work done was not reused for the model driven approach developed in Chapter 5 as the complexity of the model required too much time to logically model. Hence a more lightweight model was developed from available literature in line with how the original Generic Functional Model was created. However, this does not preclude the Generic Functional Model being logically modelled in the future to provide a more detailed and comprehensive basis for the developed model driven approach.

8.2 Reviewing the Research Objectives

In section 1.6 a number of objectives were set for this research. This section reviews the research conducted against the objectives and, ultimately, against the research problem originally stated in section 1.3. Each objective is listed as a separate subheading and reviewed accordingly.

8.2.1 Objective 1: Review General Decision Making

This objective was achieved in Chapter 2, section 2.2. Decision making under uncertainty was considered and two main decision making theories from economics were described, the *Expected Utility Theory* which states that people maximise utility as opposed to reward and *Prospect Theory* which characterises the general behaviour of decision makers with the terms Loss Aversion and Diminishing Sensitivity. The review of general decision making indicated that humans are innately irrational decision makers liable to be subject to a number of decision making fallacies.

8.2.2 Objective 2: Review Military Decision Making

This objective was achieved in Chapter 2, section 2.3. The role of information in military decision making was considered and it's increasingly important role in redefining how superiority over an enemy can be defined in terms of decision making. This found that military decision making is typically time constrained and highly stressful which limits human decision making and can negate any informational advantage. The history of decision making in military operations was traced and the Military Decision Making Process and Observe, Orient, Decide and Act Loop were discussed. Emerging tenets such as Network Centric Warfare (NCW) and its impact on situational awareness and command and control were presented. NCW is not limited to technology and could be considered to be an emerging military response to the Information Age. The proliferation of decision makers leads to alternatives to the classical decision making theories. These alternatives mostly emanate from descriptive studies of decision making in naturalistic environments. Naturalistic Decision Making is the description of proficient decision making strategies that emphasise recognitional as opposed to analytical processes. Klein's Recognition Primed Decision (RPD) strategy which is a leading theory of Naturalistic Decision Making was presented. The review of military decision making identified a number of existing decision making models, emerging military tenets and alternative decision making models which may better represent current and future military decision makers.

8.2.3 Objective 3: Identify a Decision Making Model to Support

This objective was achieved in Chapter 2, section 2.4. A leading theory of Naturalistic Decision Making is Klein's Recognition Primed Decision (RPD) strategy which has been selected as the most representative model of how military decision makers actually make decisions. The RPD model was identified as the decision making model that this research supported.

8.2.4 Objective 4: Identify a Practical Application for this Research

This objective was achieved in Chapter 2, section 2.5. To consider how to support effective military decision making a practical application was required. To ensure that the research would be valuable and applicable to the SEAS DTC a military mission was an obvious context to use. The majority of combat missions were deemed unsuitable due to security issues, especially with access to information. The Combat Search And Rescue (CSAR) mission was identified as the practical application as there is a quantity of literature (including United States military doctrines) available within the public domain. A review of this literature indicated that the mission involved decision making in a variety of situations at various levels of authority, making it suitable for this research. CSAR is also an approved SEAS DTC vignette for research. Whilst the overall outcome of the research was focused on CSAR, the less specific outcomes should be valid and applicable to a number of other mission types.

8.2.5 Objective 5: Develop an Architecture of an Approach for a Decision Support System

Two different approaches were developed to meet this objective. Firstly a parametric approach was developed in Chapter 3. A number of issues were identified with the parametric approach, principally the entanglement of functionality with implementation. This was addressed through the development of a model driven approach in Chapter 5. The model driven approach implemented the principles of MDA which specifically separates functionality from implementation. As the approach of choice in this research the developed model driven approach achieved this objective.

8.2.6 Objective 6: Implement the Developed Approach

Two different implementations were developed to meet this objective, one for the parametric approach as described in Chapter 4 and a second implementation for the model driven approach as discussed in Chapter 6. The implementation of both approaches was achieved using the Microsoft Excel software package with additional coding completed in the Visual Basic for Applications programming language.

8.2.7 Objective 7: Conduct a Set of Case Studies with the Implemented Approach

Two different sets of case studies were completed, the first using the implemented parametric approach in section 4.3 and the second set using the model driven approach in Chapter 7. Both sets used the same two historical CSAR case studies. These were the Seawolf CSAR in Vietnam, as described in the detailed narrative in subsection 4.3.1.1, and the rescue of Lt. Devon Jones, as described in subsection 4.3.2.1. These case studies were conducted firstly using the implemented parametric approach and secondly using the implemented model driven approach.

8.3 Reviewing the Research Problem

With all the objectives achieved it is seen that the research problem, as stated originally in section 1.3, has been solved through the demonstration of the model driven approach to the two specific case study examples. Thus it has been shown that a decision support system does exist:

There is a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission.

Solution of Research Problem

This research has shown that a model driven approach addresses this research problem. The analysis of a system of systems as part of a larger whole is achieved through the MDA approach of a CIM, PIM and PSM. These models separate the requirements from the functionality from the implementation. Hence it allows the larger whole (the mission itself) to be used to generate a SoS profile for analysis against the commanders intent. The mission viewpoint provides the open viewpoint while the commander's intent provides a closed viewpoint. The transformation between PIM and PSM allows rapid model generation to support the RPD model of decision making. The RPD model was identified by this research as the most representative model of how military decision makers actually make decisions. The overall approach allows a particular military mission to be captured as a model through an approach which starts with logical modelling and ends with a CIM and PIM, as described previously in section 5.5. In summary this research has shown that there *is* a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission.

8.4 Significance of Research

This section outlines the significance of the research reported in this thesis to the wider academic community. Three main areas of contribution have been identified, each of which draws from different parts of this thesis. The first area is the identification of limitations of a traditional systems engineering approach as discussed in subsection 8.4.1. The second area is the identification of a need for both open and closed viewpoints on a wicked problem as discussed in subsection 8.4.2. Finally, the third area is the demonstration of a practicable model driven approach as discussed in subsection 8.4.3.

8.4.1 Limitations of a Traditional Systems Engineering Approach

Two distinct approaches were developed in this research for a decision support system. The first approach was a parametric approach and was representative of a traditional systems engineering approach. A number of limitations were identified with this approach, some at the time of evaluation and many more when compared to the later developed model driven approach. These limitations are discussed in turn in this subsection.

The primary limitation of the parametric approach which necessitated the development of a model driven approach was the entanglement of functionality with systems. Each systems profile captured the functionality that the system could perform. This functionality then became intimately linked with the underlying platform and caused difficulty in separating them for the assessment of alternate SoS configurations.

The parametric approach did not capture any relationships between systems meaning that the combining of systems into SoS was understood as only an additive rather than an integrating function.

Overall the parametric approach was found in practice to be verbose, overly detailed and time consuming. In essence it resembled the Military Decision Making Process (MDMP), as previously described in subsection 2.3.2, more than the agile, faster Recognition Primed Decision (RPD) decision making model described in section 2.4 that this research sought.

8.4.2 Need for Closed and Open Viewpoints on a Wicked Problem

This research has established the need to take both open and closed viewpoints on a wicked problem. A closed viewpoint is not well suited for dealing with uncertainty and therefore inappropriate for wicked problems. An open viewpoint can be used to address the uncertainty of wicked problems.

The CSAR mission viewpoint is open. By association the commander's intent is open. However, the commander's RPD model, which is a proposed solution, is necessarily from a closed viewpoint. As a human, the decision making process which leads to the solution is prone to error; see the decision making fallacies described in section 2.2 and the examples given in Appendix C. It was shown from the perspective of selecting SoS to conduct CSAR missions that if only an open viewpoint is taken then the structure of the SoS to be utilised will be all the possible connections implied by the models/transforms from the open perspective. The closed viewpoint captures the structure of the SoS from an internal perspective and may be a superset or subset of the open viewpoint but can be compared and moderated by the open viewpoint. Without the open viewpoint there is no comparison and hence no indication of how appropriate the proposed SoS actually is.

The need established for both an open and closed viewpoint requires a tighter coupling of the commander's intent with the mission viewpoint. The model driven approach developed in this research achieves this through model transformation which preserves the structure and relationships of models.

8.4.3 Demonstration of a Practicable Model Driven Approach

The model driven approach developed in this research has been demonstrated as practicable through two case studies. This approach has been shown to work through the chosen epistemology of critical rationalism, i.e. the approach cannot be *proven* but rather can be shown not to be *falsified*. This research has demonstrated the model driven approach through two case studies which did not result in a falsification of the approach.

The model driven approach has been shown to handle the particular wicked problem of CSAR missions. This approach addresses both open and closed viewpoint from a bottom up and top down approach through MDA model levels. These models separate the functionality from the implementation. Hence it allows the larger whole (the mission itself) to be used to generate a SoS profile for analysis against the commanders intent. The mission viewpoint provides the open viewpoint while the commander's intent provides a closed viewpoint. The transformation between PIM and PSM provides tighter coupling between the mission viewpoint and commander's intent viewpoint and allows rapid model generation to support the RPD model of decision making.

The combination of a model driven approach with the RPD model provides the decision maker agility through a support system that is compatible with their own decision making process while informing them of the suitability of their choice compared to the mission viewpoint profile.

8.5 Further Work

Three main opportunities for further work have been identified both for applying this research and for technical extensions of the architectural approach.

Firstly, the research conducted in this thesis it should be possible to develop a tool that could support the end user decision makers. While this research was focused on the CSAR mission it could be applied to other mission types to provide support to the decision makers who must assemble system of systems to achieve their objectives.

As discussed in section 7.5 my belief is that I have tamed some elements of the CSAR wicked problem by representing elements of it in an objective manner. This is a prime area for extending this research, particularly as the GFM developed in the original parametric approach can be reused. As part of this extension the reuse of the primary and secondary attributes and factors developed for the parametric approach can also be investigated. It is envisioned at this early stage that these would be of use in differentiating between functionally identical SoS through non-functional profiling of the commanders intent.

The third opportunity identified for further work is the use of executable architectures. Executable architectures are simply architectures which contain some lower level model elements that allow them to be simulated. This in turn allows the architecture to be tested at a conceptual level to ensure it behaves as intended. For the practical application of this research it would be useful to be able to simulate the chosen architecture to help with the imagine action stage of the RPD model. To explain what executable systems architecture is it is worth reiterating what systems architecture is:

[redacted]

(Object Management Group 2007)

This specification of the parts and connectors of the system and the rules for the interactions of the parts using the connectors is realised through the use of models. In an executable architecture these models have been created utilising a standardised specification language that can be run as a program through the use of specialist simulator software. Whilst UML and MDA have been specified by the Object Management Group (OMG), there is no current specification for executable architectures and the specification language underlying them. Due to the unavailability of a standard for executable architectures a case study using a custom solution by Kennedy Carter is presented in Appendix O. This case study demonstrates model reuse through architectural patterns which is highly applicable to CSAR missions where the same conceptual model (as exposed through the development of the GFM) needs to be implemented with different systems for different applications. The architectural translation is detailed in Appendix P.

8.6 Conclusion

This thesis has developed and demonstrated a novel approach for a decision support system that can be used to analyse a system of systems as part of a larger whole from both open and closed viewpoints in order to support the decision of which systems to use to conduct a particular military mission. The view of problem solving taken in this thesis is most like the approach used by John von Neumann which he himself describes below:

[redacted]

The Mathematician by John von Neumann (von Neumann 1947, 196)

The conclusion to this entire research is thus: wicked problems do not have an exhaustively describable set of potential solutions, but that is not to say they are *unapproachable*. To tackle wicked problems requires an approach that is *elegant in its 'architectural'* and *structural makeup*. I believe that this research has developed such an approach, one that combines the generally closed viewpoint of engineering with an open viewpoint as required by wicked problems. The model driven approach developed in this research encapsulates the characteristics of the wicked problem while delivering *a few simple guiding motivations to reduce the apparent arbitrariness* of the particular situation of the problem and in so doing brings simplicity to apparent complexity.

Chapter 9

Glossary

ABCCC	Airborne Battlefield Command and Control Centre
ARRS	Aerospace Rescue and Recovery Service
ASR	Air Seas Rescue
AWACS	Airborne Warning And Control System
C2	Command and Control
CEO	Chief Executive Officer
CIM	Computational Independent Model
COA	Course Of Action
CSAR	Combat Search And Rescue
DoD	(United States) Department of Defence
DSS	Decision Support System
DTC	Defence Technology Centre
ETA	Estimated Time of Arrival
FFD	Functional Flow Diagram
GFM	Generic Functional Model
GST	General Systems Theory
IEEE	Institute of Electrical and Electronics Engineers
INCOSE	International Council On Systems Engineering
INS	Inertial Navigation System
JSRC	Joint Search and Rescue Centre
MDA	Model Driven Architecture
MDA	Model Driven Architecture
MDMP	Military Decision Making Process
MoD	Ministry of Defence
MOOTW	Military Operations Other Than War
NCW	Network Centric Warfare
NEC	Networked Enabled Capability
OMG	Object Management Group
OODA	Observe, Orient, Decide, Act
PIM	Platform Independent Model

PM	Platform Model
PSM	Platform Specific Model
RADAR	Radio Detection And Ranging
RCC	Rescue Coordination Centre
RESCAP	Rescue Combat Air Patrol
RPD	Recognition Primed Decision
S/AS	(Semi) Autonomous Systems
SAM	Surface to Air Missile
SAR	Search And Rescue
SEAS	Systems Engineering for Autonomous Systems
SFMT	Soft Factors Modelling Tool
SoS	System of Systems
UAV	Unmanned Autonomous Vehicle
UHF	Ultra High Frequency
UK	United Kingdom
UML	Unified Modelling Language
US	United States
USAF	United States Air Force
VPA	Viewpoint Analysis

Chapter 10
References

Abel, A., and S. Sukkarieh. "The coordination of multiple autonomous systems using information theoretic political science voting models." *Proceedings of the 2006 IEEE/SMC International Conference on System of Systems Engineering.* Los Angeles, CA: IEEE, 2006. 149-154.

Ackoff, R. L. "The systems revolution." *Long Range Planning* 7 (1974): 2-20.

Alberts, D. S., and R. E. Hayes. *Power to the edge.* CCRP, 2003.

Alberts, David S., John J. Garstka, and Frederick P. Stein. *Network Centric Warfare: Developing and leveraging information superiority.* 2nd Edition. Vienna, Virginia: CCRP, 1999.

Alberts, David S., John J. Garstka, Richard E. Hayes, and David A. Signori. *Understanding Information Age Warfare.* Vienna, Virginia: CCRP Publication Series, 2001.

Ancker III, C. J., and M. D. Burke. "Doctrine for Asymmetric Warfare." *Military Review*, July-August 2003: 18-25.

Bandler, R., and J. Grinder. *The structure of magic.* Santa Clara, California: Penguin, 1975.

Banker, R., and R. J. Kauffman. "The evoluation of research on information systems: A fiftieth-year survey of literature in Management Scence." *Management Science* 50, no. 3 (2004): 281-298.

Bartley, W. W. "Rationality versus the theory of rationality." In *The critical approach to science and philosophy*, by M. Bunge, 3-31. London: Transaction Publishers, 1964.

Bernoulli, D. "Exposition on a new theory on the measurement of risk." *Econometrica* 22, no. 1 (1954): 23-36.

Bernstein, P. L. *Against the gods : the remarkable story of risk.* New York: John Wiley & Sons, Inc., 1996.

Boardman, John, and Brian Sauser. "System of systems - the meaning of of." *Proceedings of rht e2006 IEEE/SMC Internation Conference on System of Systems Engineering.* Los Angeles, CA: IEEE, 2006. 118-123.

Boulding, K. E. “General systems theory - the skeleton of science.” *Management Science* 2 (1956): 197-208.

Boulding, K. E. “General systems theory: the skeleton of science.” *Emergence: Complexity & Organization (E:CO)* 6, no. 1-2 (2004): 127-139.

Boyd, J. R. “Patterns Of Conflict.” *Defense And The National Interest.* 1986. http://www.d-n-i.net/boyd/pdf/poc.pdf (accessed February 22, 2007).

—. “The Essence of Winning & Losing.” *Chet Richards.* 28 June 1995. http://www.chetrichards.com/modern_business_strategy/boyd/essence/eowl_frameset.htm (accessed May 5, 2007).

Brill, James H. “Systems engineering - a retrospective view.” *Systems Engineering* 1 (1998): 258-266.

Brooks, F. P. *The mythical man month and other essays on software engineering* . Addison-Wesley Longman, Inc., 1995.

Brown, D. *Tricks Of The Mind.* London: Transworld Publishers, 2006.

Burge, S., and M. A. Woodhead. *Functional Modelling to Perform Functional Behavioural Analysis.* Post Graduate Course Material, Loughborough: Loughborough University, 2006.

Burge, S., and M. A. Woodhead. *Viewpoint Analysis Functional Determination.* Post Graduate Course Material, Loughborough: Loughborough University, 2006.

Bushey, D. A., and M. J. Forsyth. “The recognition primed decision model: an alternative to the MDMP for GWOT.” *Field Artillery*, January-February 2006: 10-13.

Calderwood, R., B. W. Crandall, and G. A. Klein. *Expert and novice fire ground command decisions.* Interim Report KA-TR-858(D)-87-03F, Alexandria, VA: U.S. Army Research Institute for the Bahioral and Social Sciences, 1987.

Chen, P., and J. Clothier. “Advancing systems engineering for systems-of-systems challenges.” *Systems Engineering* 6, no. 3 (2003): 170-183.

Chomsky, N. *Syntactic structures.* Berlin: Walter de Gruyter GmbH & Co., 1957.

—. *Topics in the Theory of Generative Grammar.* The Hague, The Netherlands: Moulton & Co. N.V., 1966.

Cleveley, J., P. Johnson, C. E. Siemieniuch, and M. Woodhead. *SEIC-RP-0593 Decision Support System.* SEAS DTC Technical Report, SEAS DTC, 2007.

Cleveley, J., P. Johnson, C. E. Siemieniuch, and M. Woodhead. *SEIC-RP-0595 Generic Functional Model for Combat Search and Rescue (CSAR).* SEAS DTC Report, SEAS DTC, 2007.

Cook, S. C. "On the acquisition of systems of systems." *Proceedings of the INCOSE 2001 Annual Symposium.* Melbourne, Australia: INCOSE, 2001.

De Laurentis, Dan, et al. "A Case fpr an International Concortium on System-of-Systems Engineering." *IEEE Systems Journal* 1, no. 1 (September 2007): 68-73.

Defense Acquisition University. "Chapter 4.2.6. System of Systems Engineering." *Defense Acquisition Guidebook.* 7 October 2006. https://akss.dau.mil/DAG/Guidebook/IG_c4.2.6.asp (accessed July 6, 2009).

Department of Defense. "Joint Concept Development and Revision Plan." *Marine Corps Combat Development Command.* July 2004. https://www.mccdc.usmc.mil/OpsDiv/Integration/Intel/intelligence_integration_files/JOCs/040730jcdrp_opsdepsapproved.doc (accessed July 28, 2009).

Department Of The Army. "Field Manual 101-5 Staff Organisation And Operations." *Defence Technical Information Center.* 31 May 1997. http://www.dtic.mil/doctrine/jel/service_pubs/101_5.pdf (accessed April 30, 2007).

Dickerson, C. E., and D. N. Mavris. *Architecture and Principles of Systems Engineering.* New York: Taylor & Francis LLC, 2009.

Dickerson, C. "Towards a Logical and Scientific Foundation for System Concepts, Principles, and Termnology." *3rd IEEE International Symposium on Service-Oriented Systems Engineering.* Monterey, CA: IEEE, 2008.

Evans, Andy. *Combat Search & Rescue.* London: Arms & Armour, 1999.

Farina, J. "The Monty Hall Page." *University of California, San Diego, Department of Mathematics.* http://math.ucsd.edu/~crypto/Monty/montybg.html (accessed January 15, 2008).

Ford, T. C., J. M. Colombi, D. R. Jacques, and S. R. Graham. "On the application of classification concepts to systems engineering design and evaluation." *Systems Engineering* 12, no. 2 (2009): 141-154.

Francois, C. "Systemics and cybernetics in a historical perspective." *Systems Research and Behavioral Science* 16, no. 9 (1999): 203-219.

François, C. "Systemics and cybernetics in a historical perspective." *Systems Research and Behavioral Science* 16, no. 9 (1999): 203-219.

Gorod, Alex, Brian Sauser, and John Boardman. "System-of-systems engineering management: a review of modern history and a path forward." *IEEE Systems Journal* 2 (December 2008): 484-499.

Graham, J. J. "ON WAR: Book I - On the Nature of War: Chapter I - What is War?" *The Clausewitz homepage.* 26 November 2008. http://www.clausewitz.com/CWZHOME/VomKriege2/BK1ch01.html (accessed February 6, 2009).

Granberg, D., and P. R. Mueser. "The Monty Hall Dilemma Revisited: Understanding The Interaction Of Problem Definition And Decision Making." *CiteSeerX.* May 1999. http://citeseer.ist.psu.edu/289095.html (accessed January 3, 2007).

Hagee, M. W. *A Concept for Distributed Operations.* Washington, D.C.: Department of the Navy, 2005.

Harvey, J. B. "The Abilene Paradox: The Management Of Agreement." *Organizational Dynamics*, 1988: 17-43.

Hatley, Derek J, Peter Hruschka, and Imtiaz A Pirbhai. *Process for System Architecture and Requirements Engineering.* New York: Dorset House Publishing, 2000.

Heinl, Robert Debs. *Dictionary of Military and Naval Quotations.* Naval Institute Press, 1966.

Hitchins, D. K. *Advanced systems thinking, engineering and management.* Boston: Artech House, 2003.

Hodgson, A., and C. E. Siemieniuch. *Project SER006 - Impact of Different Cultural Attribute Sets on S/AS Decision Structures and Interfaces - Final Report.* Technical Report, SEAS-DTC Report No. SEIC-RP-0780, 2009.

Hodgson, A., and C.E. Siemieniuch. "A software tool for evaluating the effects of culture on military operations." *Proceedings of the AAAI Fall Symposium Program: Adaptive Agents in Cultural Contexts.* Arlington, USA: Association for the Advancement of Artificial Intelligence (AAAI), 2008.

Hubbard, E. M. *Supporting the Configuration of Decision-Making Systems for Complex, Long-Life Engineered Systems.* Ph.D. Dissertation, UK: Loughborough University, 2010.

IEEE Computer Society. *IEEE Standard 1471: Recommended Practice for Architecture Description of Software-Intensive Systems.* New York: Institute of Electrical and Electronics Engineers, 2000.

International Council on Systems Engineering. *INCOSE Systems engineering handbook, v.3.1.* Seattle: INCOSE Central Office, 2007.

Jackson, M. C., and P. Keys. "Towards a system of systems methodologies." *Journal of the Operational Research Society* 35, no. 6 (1984): 473-486.

Jamshidi, Mo. "System-of-systems engineering - a definition." *IEEE SMC.* Piscataway, NJ: IEEE, 2005.

Janis, I. L. *Groupthink.* Boston: Houghton Mifflin Company, 1982.

Johnson, P. W., C. E. Siemieniuch, and M. A. Woodhead. "A decision making support system utilising the evaluation of whole system capability against mission requirements." *Institiute of Engineering and Technology 2nd Autonomous SYstems Conference.* London, UK: Institute of Engineering and Technology (IET), 2007c. Paper reference 009.

—. "A method for evaluating appropriate system architecture options for generic and specific mission requirements." *2nd Annual SEAS DTC Conference.* Edinburgh, UK: Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC), 2007b. Paper reference SER006.

—. "An initial set of case studies utilising a developed decision support system for evaluating available systems against mission requirements." *3rd Annual SEAS DTC Conference.* Edinburgh, UK: Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC), 2008. Paper reference SER006.

—. "Identifying and assessing appropriate system architecture options for generic and specific mission requirements." *12th ICCRTS Conference.* Newport, Rhode Island, USA: International Command and Control Research and Technology Symposia (ICCRTS), 2007a. Paper Reference 203.

Joint Chiefs of Staff. *Joint Publication 3-50.2 Doctrine for Joint Combat Seach and Rescue.* Doctrine, Fort Belvoir: Defence Technical Information Centre, 1996.

Joint Chiefs of Staff. *Joint Publication 3-50.21 Joint Tactics, Techniques and Procedures for Combat Search and Rescue.* Doctrine, Fort Belvoir: Defense Technical Information Centre, 1998.

—. *Joint Vision 2020.* Washington, DC: US Government Printing Office, 2000.

Jones, V., A. van Halteren, D. Konstantas, I. Widya, and R. Bults. "An application of augmented MDA for the extended healthcare enterprise." *International Journal of Business Process Integration and Management* 2, no. 3 (2007): 215-229.

Kahneman, D., and A. Tversky. "Prospect theory: an analysis of decision under risk." *Econometrica* 47, no. 2 (March 1979): 263-292.

Kelly, John Larry. "A New Interpretation of Information Rate." *The Bell System Technical Journal*, July 1956: 917-926.

Klein, G. A., and R. Calderwood. "Decision models: some lessons from the field." *IEEE Transactions on Systems, Man and Cybernetics* 21, no. 5 (1991): 1018-1026.

Klein, G. A., R. Calderwood, and D. MacGregor. "Critical decision method for eliciting knowledge." *IEEE Transactions on Systems, Man, and Cybernetics* 19, no. 3 (1989): 462-472.

Klein, G. *Sources Of Power: How People Make Decisions.* Cambridge: The MIT Press, 1999.

Klein, G., and B. Crandall. *Recognition primed decision strategies.* Final Project Report - ARI Research Note 96-36, Alexandria, VA: United States Army Research Institute for the Behavioral and Social Sciences, 1996.

Kottek, M., J, Grieser, C. Beck, B. Rudolf, and F. Rubel. "World Map of the Köppen-Geiger Climate Classification." *Meteorologische Zeitschrift* 15, no. 3 (June 2006): 259-263.

Maier, M. W. "Architecting Principles for Systems of Systems." *Proceedings of the Sixth Annual International Symposium, INCOSE.* Boston: International Council on Systems Engineering (INCOSE), 1996.

Maier, M. W., and E. Rechtin. *The Art Of Systems Architecting.* 2nd Edition. Boca Raton: CRC Press LLC, 2000.

McLamb, J. S. "Is it time to abandon the military decision making process?" *Military Review*, 2002: 88-102.

McManus, H., M. Richards, A. Ross, and D. Hastings. "A framework for incorporating "ilities" in tradespace studies." *AIAA Space 2007.* Long Beach, C: American Institute of Aeronautics and Astronautics, 2007.

Miller, Joaquin, and Jishnu Mukerji. "MDA Guide Version 1.0.1." *Object Management Group.* 12 June 2003. http://www.omg.org/docs/omg/03-06-01.pdf (accessed March 3, 2009).

Ministry of Defence. "What is Through Life Capability Management ?" Vers. 2.0.14. *Acquisition Operating Framework.* July 2009. http://www.aof.mod.uk/aofcontent/operational/business/capabilitymanagement/capabilitymanagement_whatis.htm (accessed July 24, 2009).

Morgan, G. F. *Realigning Joint Doctrine For Effective Theater Combat Search And Rescue (CSAR) Operations.* Dissertation, Newport, RI: Joint Military Operations Department, Naval War College, 2003.

National Museum of the USAF. "Desert Storm Rescue." *National Museum of the USAF Web site.* http://www.nationalmuseum.af.mil/factsheets/factsheet.asp?id=1250 (accessed December 5, 2008).

National Musueum of the USAF. "Air Rescue Service Established." *National Musueum of the USAF web site.* http://www.nationalmuseum.af.mil/factsheets/factsheet.asp?id=1233 (accessed December 5, 2008).

Object Management Group. "OMG Unified Modeling Language (OMG UML), Infrastructure, V2.1.2." *Object Management Group.* 4 November 2007. http://www.omg.org/spec/UML/2.1.2/Infrastructure/PDF (accessed May 1, 2008).

Office of the Deputy Under Secretary of Defence for Acquisition and Technology, Systems and Software Engineering. *Systems Engineering Guide for Systems of Systems, Version 1.0.* Washington, DC: ODUSD(A&T)SSE, 2008.

Oxford English Dictionary. 2nd Edition. Oxford: Clarendon Press, 1989.

Paparone, C. R. "US Army Decision Making: Past, Present And Future." *Military Review*, July-August 2001: 45-53.

Perrow, C. *Normal Accidents.* Princeton: Princeton University Press, 1984.

Pokrant, Marvin. *Desert Storm at Sea: What the Navy Really Did .* Westport: Greenwood Press, 1999.

Poundstone, William. *Fortune's Formula.* New York: Hill and Wang, 2005.

Raistrick, Chris, Paul Francis, John Wright, Colin Carter, and Ian Wilkie. *Model Driven Architecture with Executable UML.* Cambridge: Cambridge University Press, 2004.

Rechtin, E. *Systems Architecting Of Organizations: Why Eagles Can't Swim.* Boca Raton: CRC Press LLC, 1999.

Rhodes, D. H. "Highlights Of Systems Engineering Research." *BAE SYSTEMS SE&I Conference.* Cambridge, 2006.

Rittel, H. W. J., and M. M. Webber. "Dilemmas in a general theory of planning." *Policy Sciences* 4 (1973): 155-169.

Ross, K. G., G. A. Klein, P. Thunholm, J. F. Schmitt, and H. C. Baxter. "The recognition-primed decision model." *Military Review*, July-August 2004: 6-10.

Sage, A. P., and S. M. Biemer. "Processes for system family architecting, design and integration." *IEEE Systems Journal* 1, no. 1 (September 2007): 5-16.

Saunders, T., et al. *System-of-Systems Engineering for Air Force Capability Development.* Washington DC: Scientific Advisory Board (United States Air Force), 2005.

Schmitt, J., and G. Klein. "A recognitional planning model." *Proceedings of the 1999 Command and Control Research and Technology Symposium.* Newport, Rhode Island: Naval War College: ICCRTS, 1999. 510-522.

Schoemaker, P. J. "The expected utility model: its variants, purposes, evidence and limitations." *Journal of Economic Literature* XX (1982): 529-563.

SEAS DTC. *Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC) Homepage.* 11 September 2008. http://www.seasdtc.com/ (accessed December 1, 2008).

Shannon, Claude Elwood. "A Mathematical Theory of Communication." *The Bell System Technical Journal* 27 (July, October 1948): 379-423, 623-656.

Shattuck, L. G. "Communicating Intent and Imparting Presence." *Military Review*, March-April 2000: 66-72.

Siemieniuch, C. E., and N. Meese. "Impact of different cultural attribute sets on S/AS decision structures and interfaces." *Proceedings of SEAS DTC Annual Technical Conference.* Edinburgh, UK: Systems Engineering for Autonomous Systems (SEAS) Defence Technology Centre (DTC), 2006.

Siemieniuch, Carys E., and Murray A. Sinclair. "Impact of Cultural Attributes on Decision Structures and Interfaces." *Proceedings of the 11th ICCRTS Coalition Command and Control in the Networked Era.* Cambridge: The Command and Control Research Program, 2006.

Simon, Herbert. "Rational Decision Making in Business Organization (The 1978 Nobel Memorial Prize in Economics Lecture)." *Carnegie Mellon University Libraries Herbert Simon Collection.* 27 February 1979. http://diva.library.cmu.edu/webapp/simon/item.jsp?q=/box00082/fld06591/bdl0003/doc0001/ (accessed January 30, 2009).

Soley, R. M. “Leveraging Model Driven Architecture® for Model Driven Systems Engineering.” *Object Management Group, Inc. (OMG®).* March 2004. http://www.omg.org/~soley/mdase.ppt (accessed February 10, 2009).

Starmer, C. “Developments in non-expected utility theory: the hunt for a descriptive theory of choice under risk.” *Journal of Economic Literature* XXXVIII (June 2000): 332-382.

Taleb, N. N. *Fooled by randomness.* London: Penguin Books Ltd., 2007.

Taylor Jr., L B. *That Others May Live.* New York: E.P.Dutton & Co., 1967.

Thornton-Wells, T. A., J. L. Moore, and J. L. Haines. “Genetics, statistics and human disease: analytical retooling for complexity.” *TRENDS in Genetics* (Elservier) 20, no. 12 (2004): 640-647.

Thunholm, P. “A new model for tactical mission planning for the Swedish armed forces.” *2006 Command and Control Research and Technology Symposium.* San Diego, CA: The Command and Control Research Program, 2006.

Thunholm, P. “Planning under time pressure: an attempt toward a prescriptive model of military tactical decision making.” In *How professionals make decisions*, by H. Montgomery, R. Lipshitz and B. Brehmer, 43-56. Boca Raton, FL: CRC Press, 2004.

Tversky, A., and D. Kahneman. “Loss aversion in riskless choice: a reference-dependent model.” *The Quarterly Journal of Economics* 106, no. 4 (November 1991): 1039-1061.

U.S Department Of Defense. *Joint Pub 3-50.2 Doctrine for Joint Combat Search and Rescue.* Fort Belvoir: Defense Technical Information Centre, 1996.

U.S. Department Of Defense. *Joint Doctrine Encyclopedia.* Fort Belvoir: Defense Technical Information Centre, 1997.

—. *Joint Pub 3-0 Joint Operations.* Fort Belvoir: Defense Technical Information Centre, 2006.

—. *Joint Pub 3-50.21 Joint Tactics, Techniques and procedures for Combat Search and Rescue.* Fort Belvoir: Defense Technical Information Centre, 1998.

—. *Joint Pub 3-50.3 Joint Doctrine for Evasion and Recovery.* Fort Belvoir: Defense Technical Information Centre, 1996.

—. *Joint Publication 1-02 Department of Defense Dictionary of Military and Associated Terms.* Fort Belvoir: Defense Technical Information Centre, 2006.

United States Air Force. *Air Force Doctrine Document 2-1.6 Combat Search and Rescue.* Doctrine, Fort Belvoir: Defense Technical Information Centre, 1998.

von Bertalanffy, L. “The theory of open systems in physics and biology.” *Science* 111 (January 1950): 23-29.

von Neumann, J. “The Mathematician.” In *Works of the mind*, by R. B. Heywood, 180-196. Chicago: University of Chicago Press, 1947.

Widder, W. “Auftragstaktik and Innere Führung: Trademarks of German Leadership.” *Military Review*, September-October 2002: 3-9.

Wu, G., J. Zhang, and R. Gonzalez. “Decision under risk.” In *Blackwell handbook of judgement and decision making*, by D. K. Koehler and N. Harvey, 399-423. Hoboken, NJ: Wiley-Blackwell, 2004.

Yourdon, E. *Modern strucutred analysis.* Upper Saddle River, NJ, USA: Yourdon Press, 1988.

Chapter 11

Appendices

Appendix A

Wicked Problems

Rittel and Webber (1973) specify ten characteristics of wicked problems:

[redacted]

Appendix B

Characterising System-of-Systems

There is currently no universally accepted definition for system of systems (SoS). A number of authors have proposed characterisations for systems of systems.

Maier states the following five characteristics that differentiate a system of systems from a system.

[redacted]

Maier (Maier 1996)

Boardman and Sauser differentiated between systems and system of systems as presented below in Table 11.

Table 11 – Differentiating a System from a System of Systems (Boardman and Sauser 2006, 121)

Element	System	System of Systems
xxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxx
xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxx
xxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxx
xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxx
xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxx	xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxxxxxxxx

Appendix C
Decision Making Examples

To illustrate human fallacies with respect to decision making a couple of famous puzzles are presented in this Appendix. The first problem, the Monty Hall Problem examines vertical paradoxes in individual decision making. The second problem examines a phenomenon of groupthink, where group members go along with what they *believe* is the consensus.

C.1 The Monty Hall Problem

It has been suggested that this problem first appeared in Joseph Bertrand's *Calcul des probabilités* published in 1888 and is now probably best known as the Monty Hall Problem. Monty Hall was the presenter of an American TV show called Let's Make A Deal which was used as the setting for more modern retellings of the problem. Whilst the scenario changes the structure and outcome of the problem are the same. For clarity and eloquence the scenario presented here is the version preferred by the psychological illusionist and showman Derren Brown.

[redacted]

(Brown 2006, 261-262)

This is the most simple of decisions to make as there are only two clear courses of action to choose between. The reader is encouraged to make their own decision now.

The most common answer is that an individual would stick. Many reasons are quoted, but generally it is because it seems *right* and *comfortable* to stick because of the tacit assumptions already made (Granberg and Mueser 1999, 32). The correct answer is that you should switch, every time. The probability of initially picking the box with the ring in it is $\frac{1}{3}$, so you have a probability of $\frac{2}{3}$ of having your finger on an empty box. An empty box is removed, leaving just the two. One of these boxes has the ring in it and two out of three times it'll be the box that you have not got your finger on.

This problem was posed to Marilyn Vos Savant's Sunday Parade column and she gave the answer as above which attracted nearly 10,000 Reponses from readers often vehemently disagreeing with her (Farina n.d.). The reason why you should switch is based on the fact, as given in Brown's example, that the host knows which box the ring is in.

Whilst a good brain teaser and an excellent way of annoying friends and strangers alike this does show a great fallacy with human decision making when presented with such cognitive traps. Such predispositions to erroneously understandings of probability means that people, with the best of intentions and in simple either/or decision situations, can make the wrong decision.

As a side note, and pure conjecture, I wonder if part of the reason for sticking with the ring box you originally selected is because you have already made the decision and hence no further decision is required. By selecting a ring box you have already decided which box has the ring in it and, although another box has been removed, it does not alter the state of the box with your finger on it which you believe contains the ring. To switch your selection of which box to select would be to show that your first decision was incorrect. As your original decision has not been proved wrong yet what is the point in changing you mind when you may have been right?

Having considered an example of individual decision making let us now consider decision making by a group of decision makers.

C.2 The Abilene Paradox

This refers to an anecdote in a paper written by the management expert Jerry Harvey (Harvey 1988). It is an example of groupthink, which is where group members conform to what they believe is the group's consensus rather than their own opinions. An abbreviated anecdote of the Abilene Paradox is quoted below :

[redacted]

[redacted]

As Harvey stated in his paper, "The inability to manage agreement, not the inability to manage conflict, is the essential symptom that defines organizations caught in the web of the Abilene Paradox" (Harvey 1988, 66). Groupthink has been blamed for a number of disasters, perhaps most famously the Challenger Space Shuttle disaster. Janis developed a set of eight symptoms to test groupthink (Janis 1982):

[redacted]

Groupthink (Janis 1982, 174) (abbreviated)

Appendix D

The Military Decision Making Process

The purpose of this Appendix is to present the details of the Military Decision Making Process (MDMP) and in so doing illustrate its complexity and time consuming nature.

D.1 Step 1 - Receipt of Mission.

[redacted]

[redacted]

D.2 Step 2 - Mission Analysis.

[redacted]

[redacted]

D.3 Step 3 - Course of Action Development.

[redacted]

[redacted]

D.4 Step 4 - Course of Action Analysis.

[redacted]

[redacted]

D.5 Step 5 - Course of Action Comparison.

[redacted]

D.6 Step 6 - Course of Action Approval.

[redacted]

D.7 Step 7 - Orders Production.

[redacted]

Appendix E

CSAR View Point Analysis

This Appendix presents the View Point Analysis diagrams produced as part of this research.

E.1 CSAR Overview

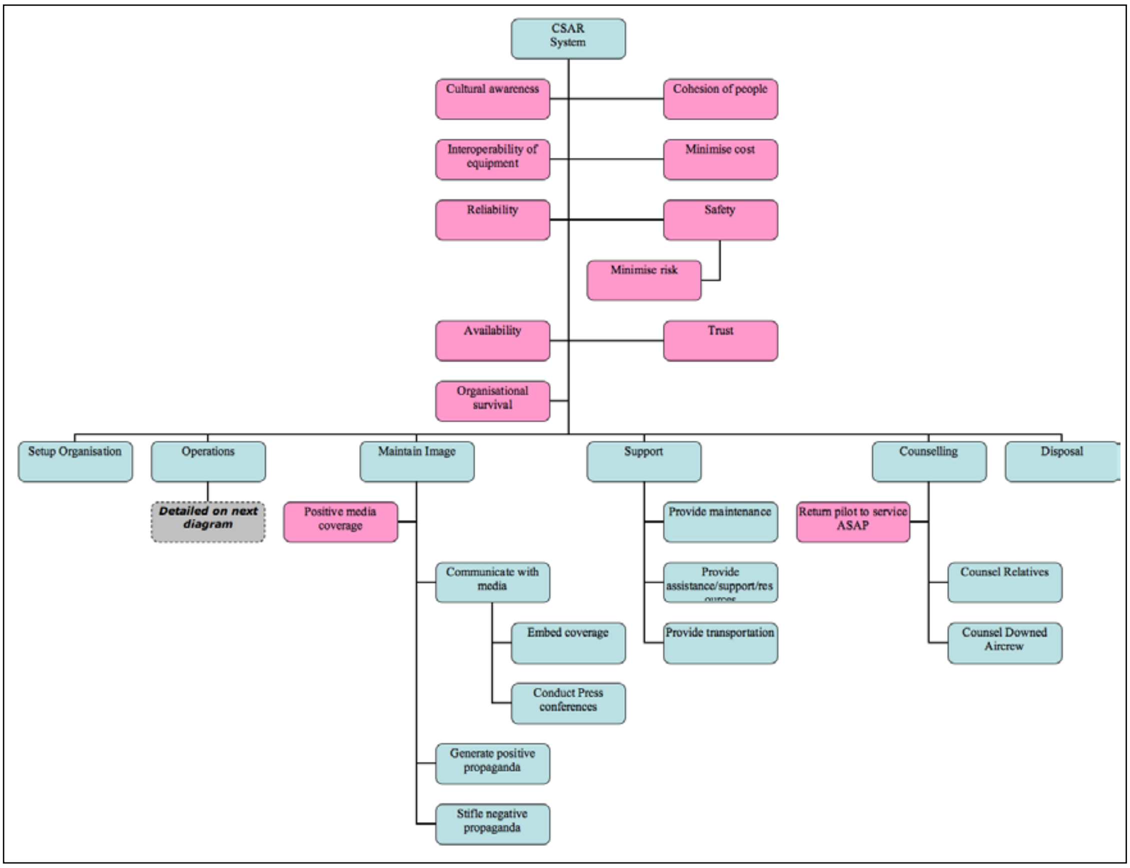


Figure 76 – CSAR Overview VPA

E.2 CSAR Operations Focus

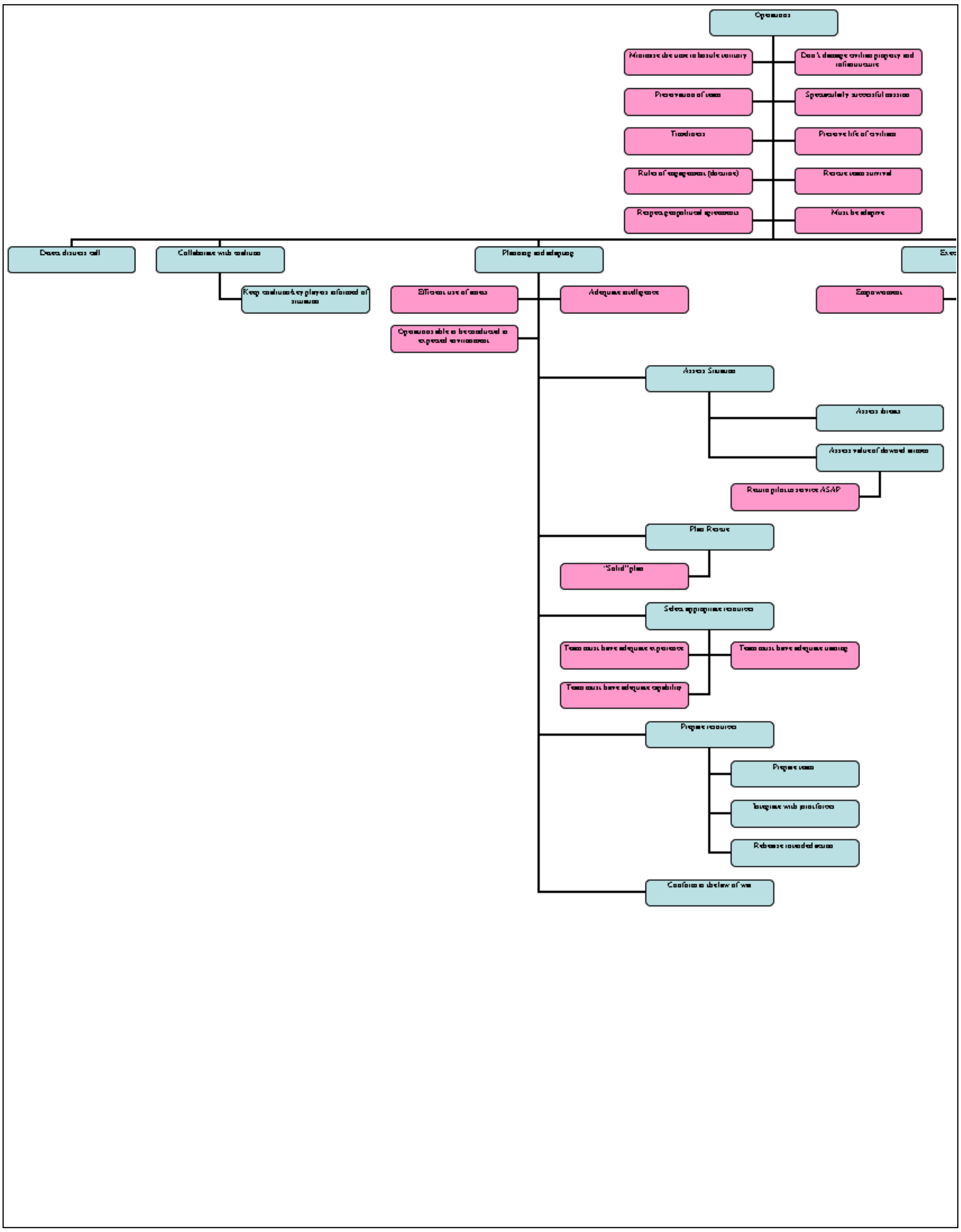


Figure 77 – CSAR Operations Focus VPA (part one)

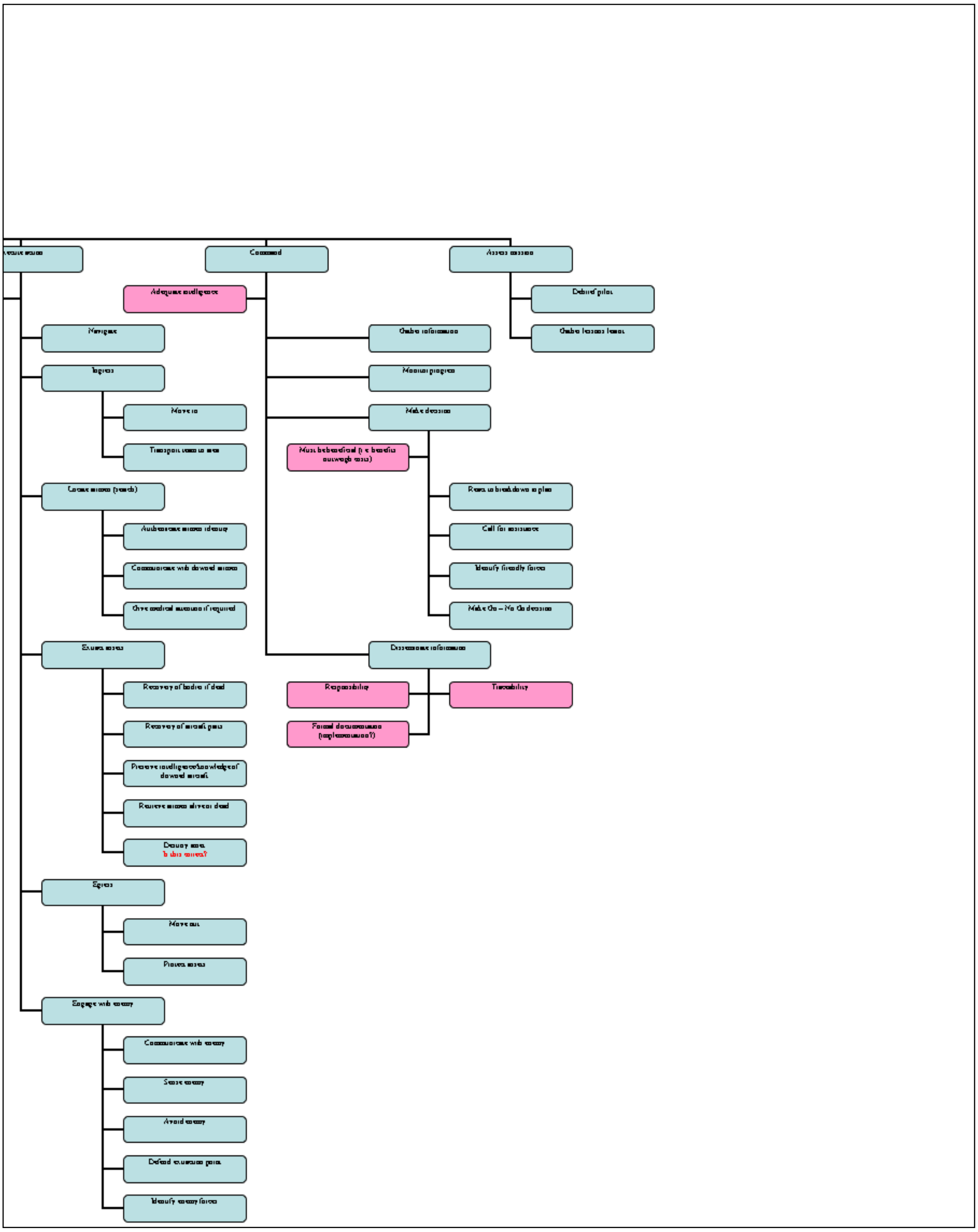


Figure 78 – CSAR Operations Focus VPA (part two)

E.3 VPA For Scenario 2

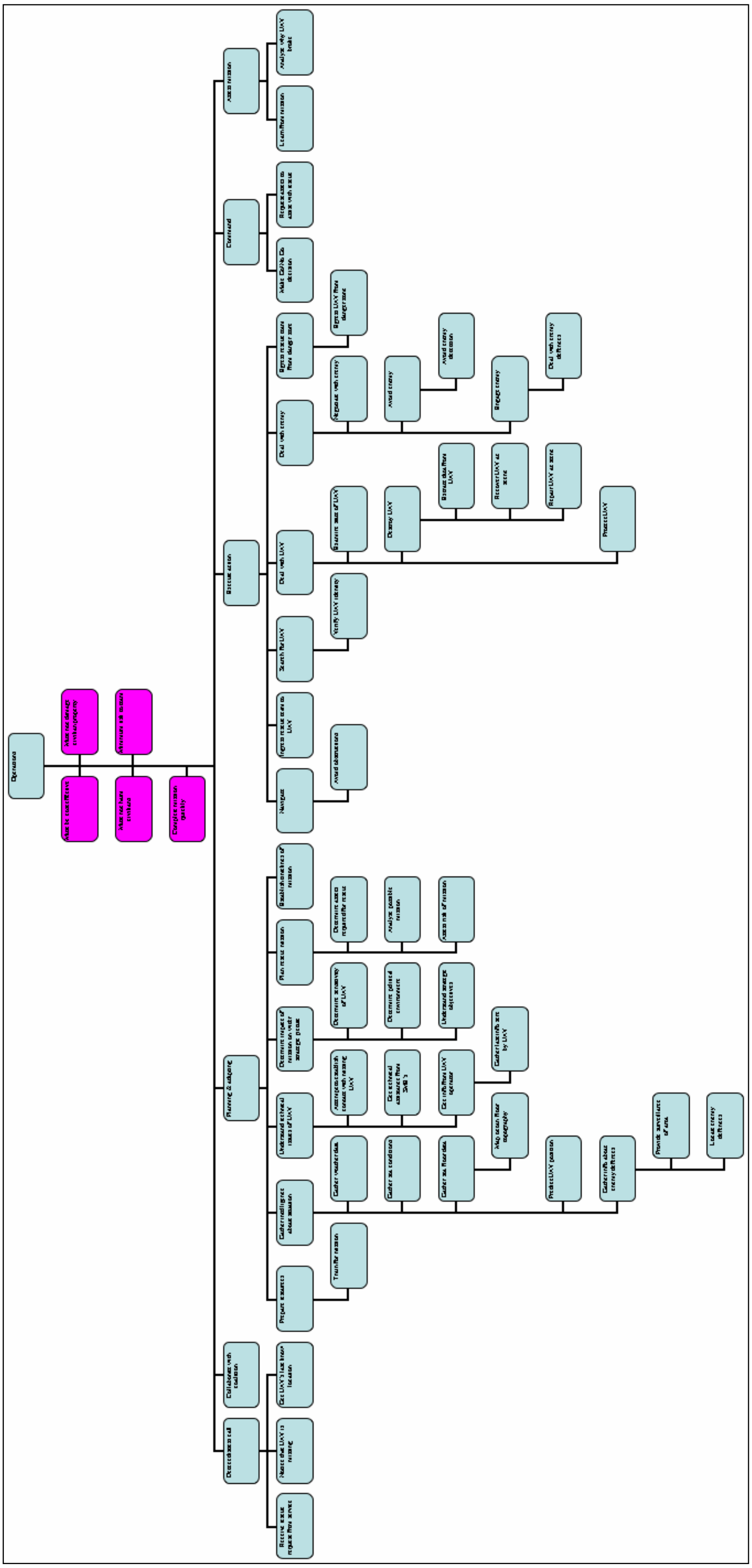


Figure 79 – Scenario 2 VPA

Appendix F

The CSAR Generic Functional Model

This section of the Appendix presents the Generic Functional Model, developed as part of this research. The Generic Functional Model aimed to capture the functionality required by the combat search and rescue mission. The full set of Functional Flow diagrams that constitute the Generic Functional model are presented in the course of this Appendix starting with the highest level Context Diagram.

F.1 The Context Diagram

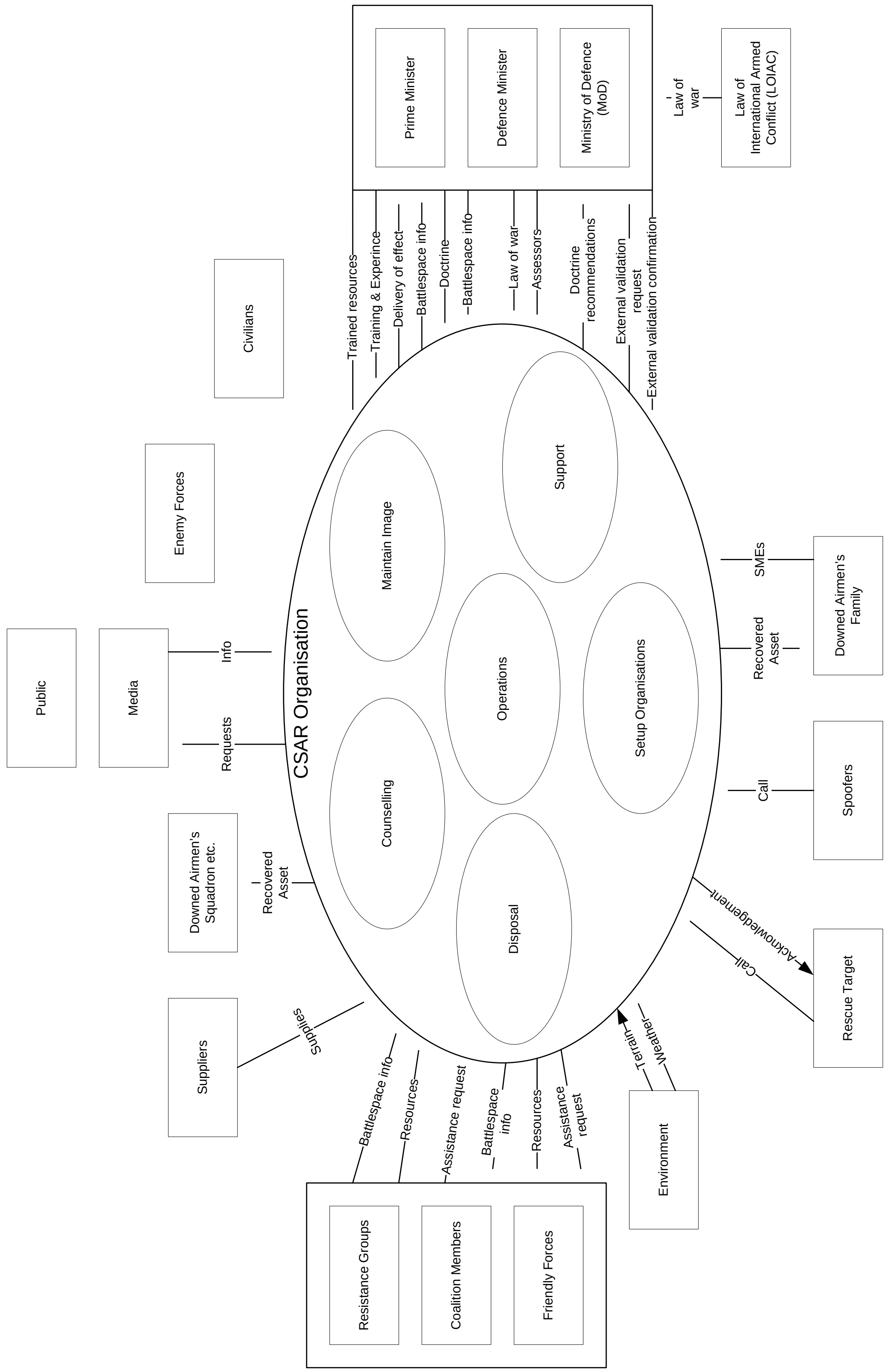


Figure 80 – Context FFD

F.2 Level 0 (Operations)

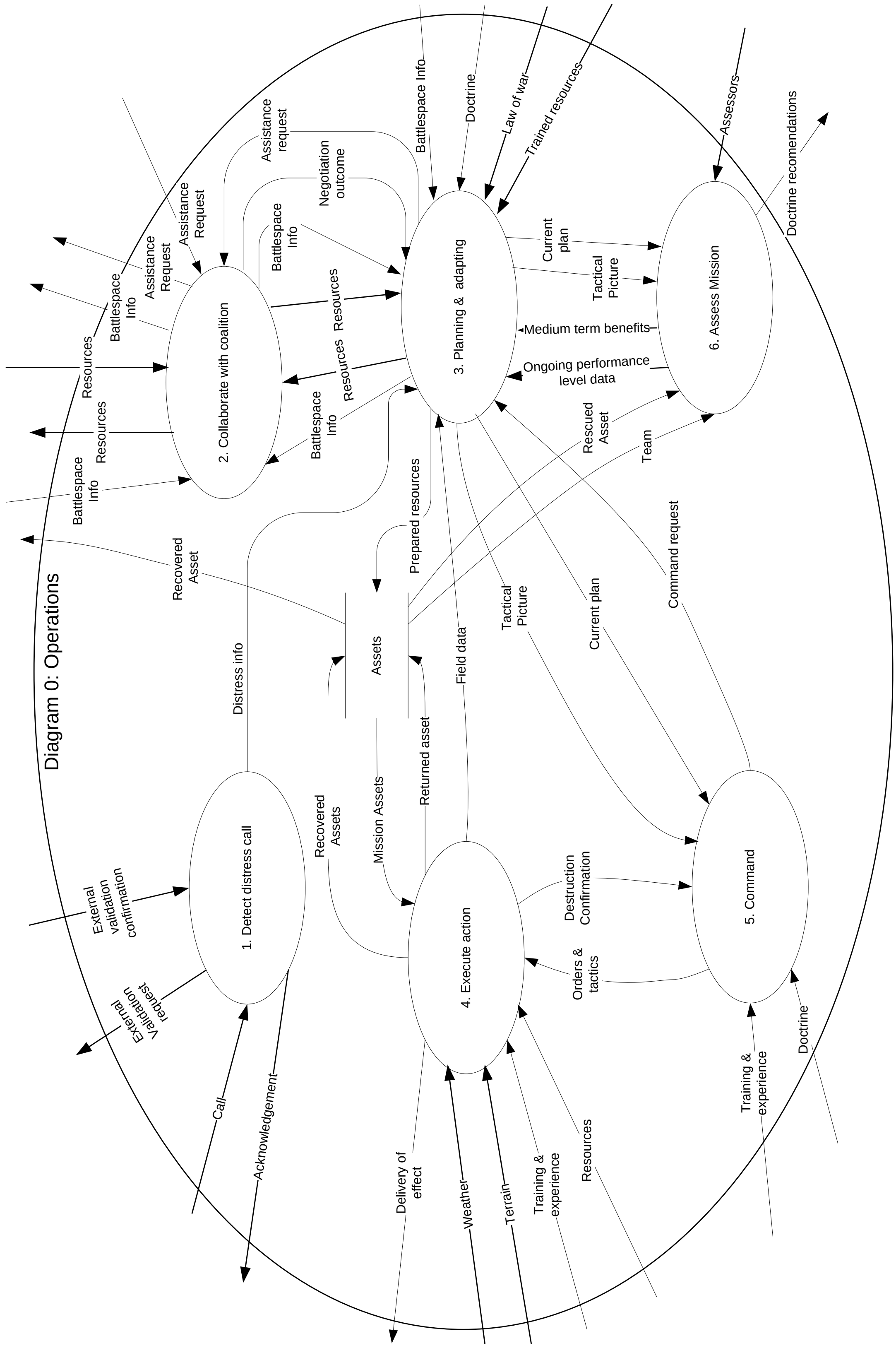


Figure 81 – Level 0 (Operations) FFD

F.3 Level 1 (Detect Distress Call)

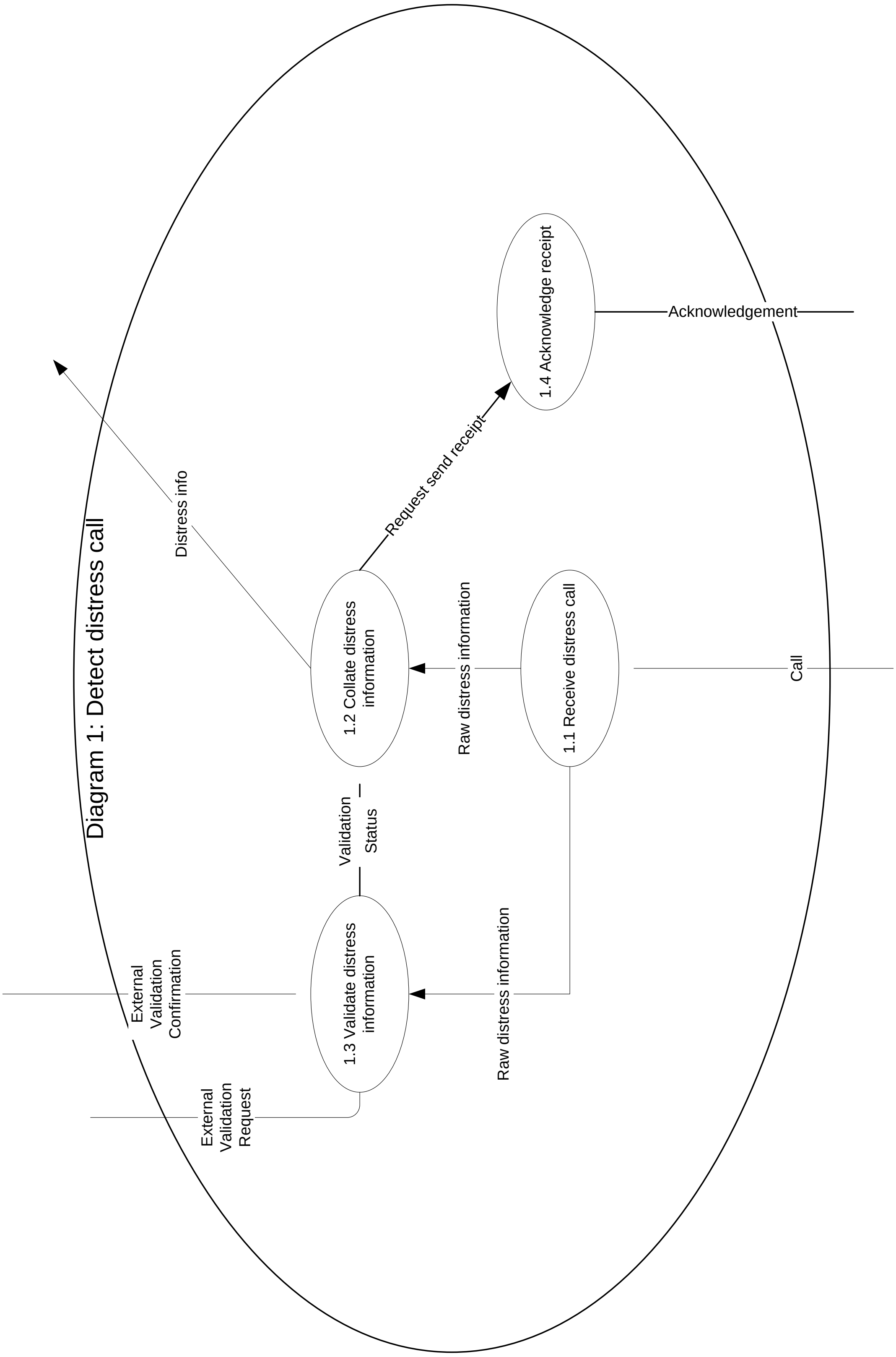


Figure 82 – Level 1 (Detect Distress Call) FFD

F.4 Level 2 (Collaborate With Coalition)

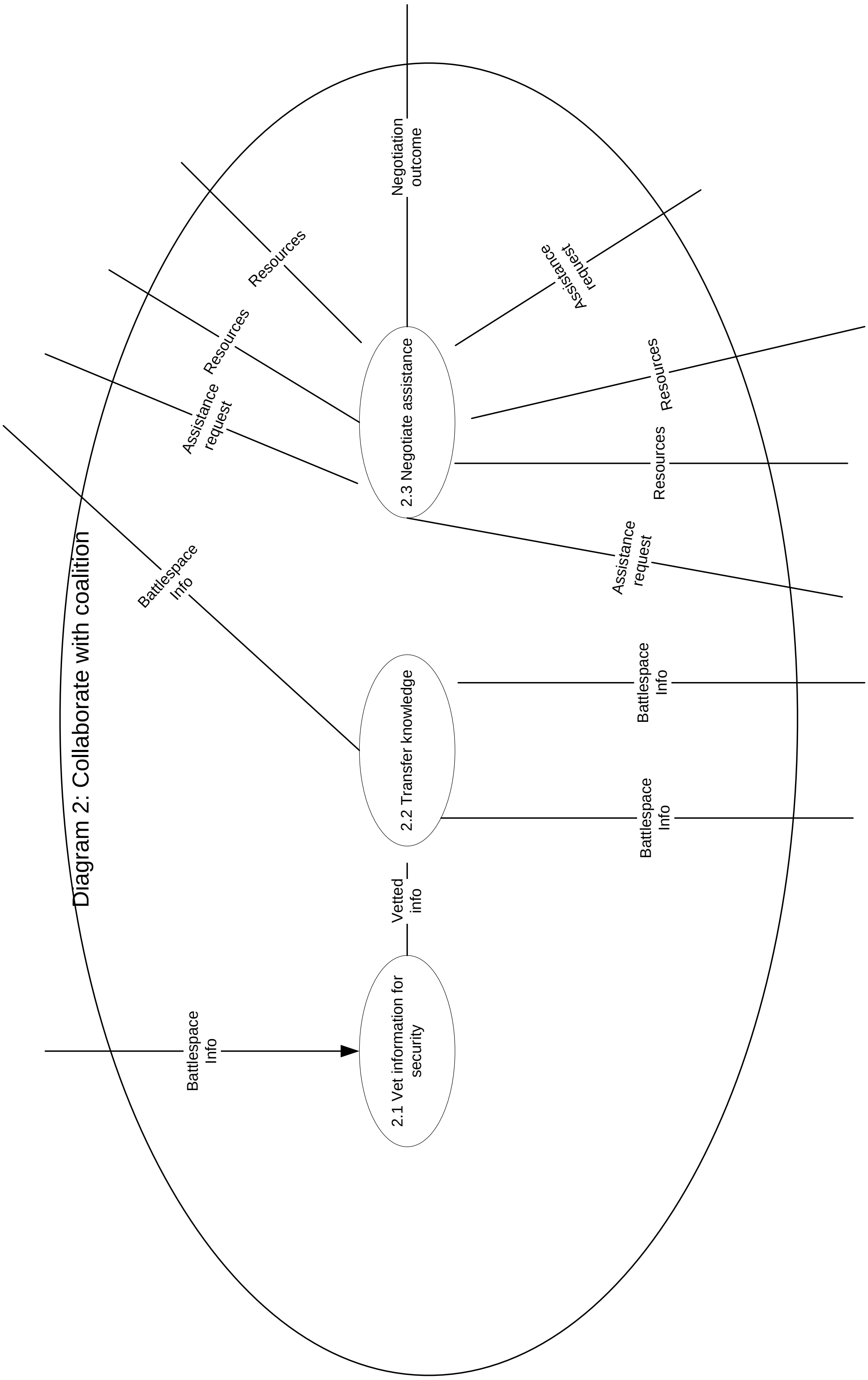


Figure 83 – Level 2 (Collaborate With Coalition) FFD

F.5 Level 3 (Planning & Adapting)

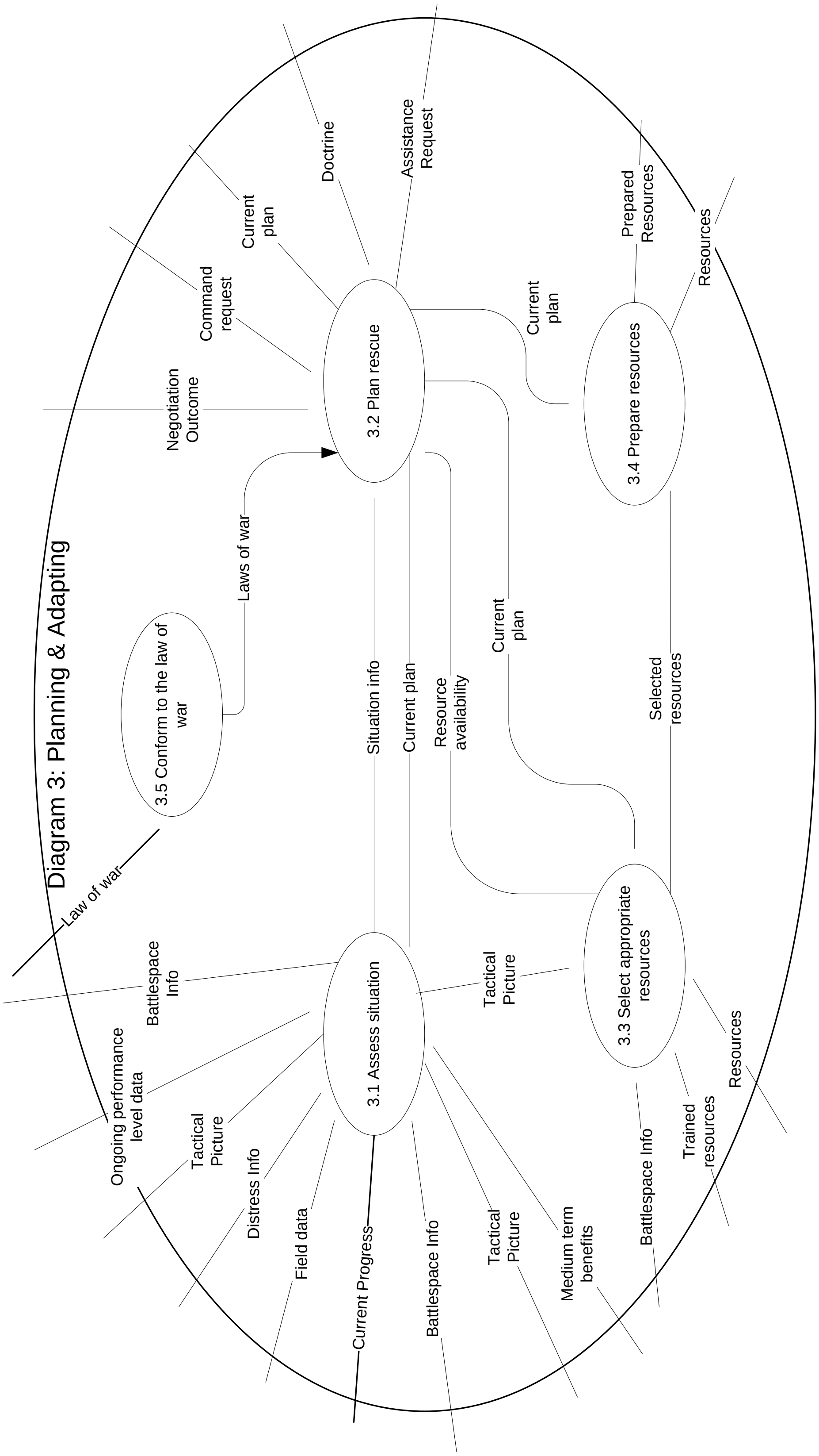


Figure 84 – Level 3 (Planning & Adapting) FFD

F.6 Level 3.1 (Assess Situation)

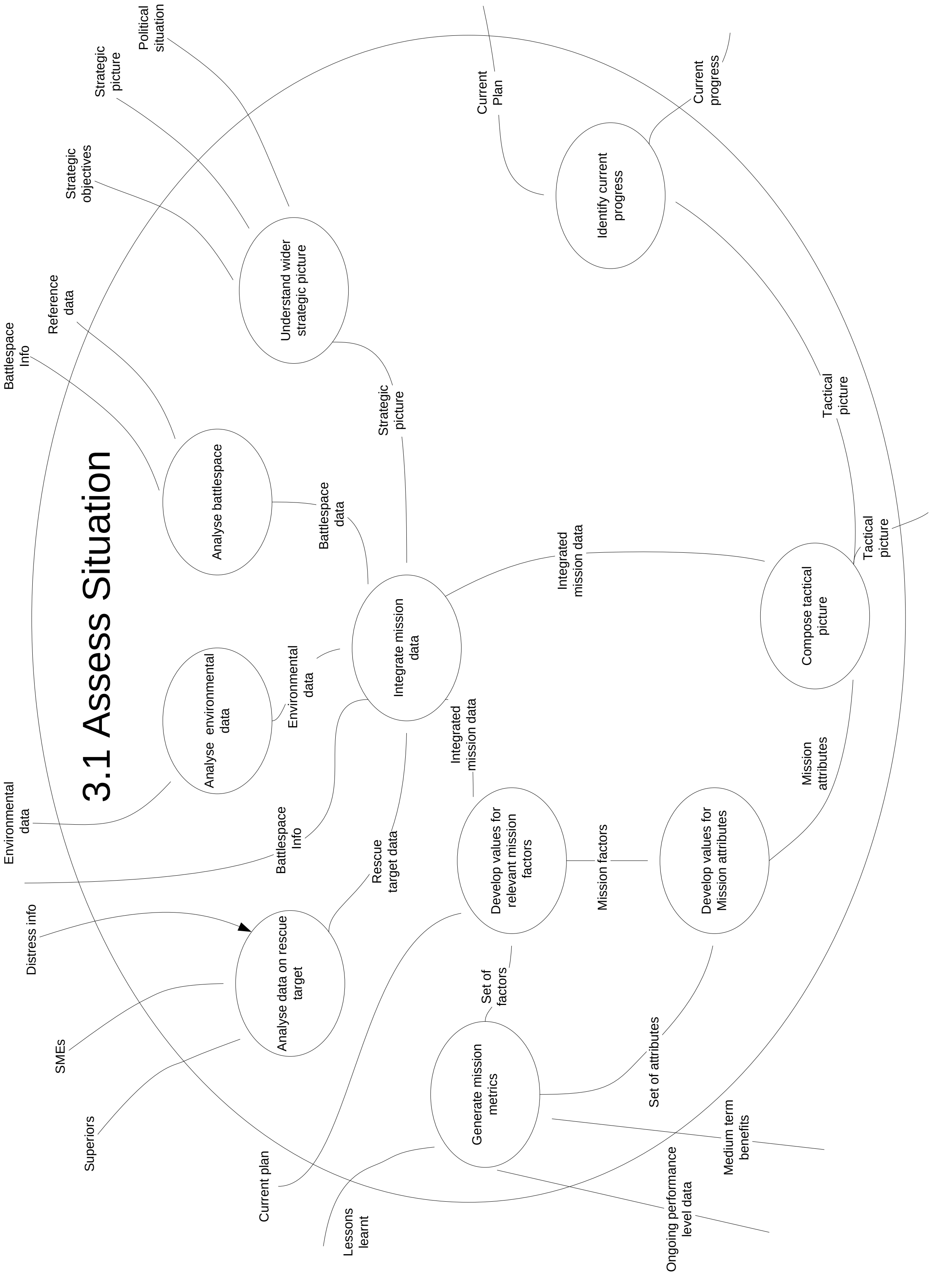


Figure 85 – Level 3.1 (Assess Situation) FFD

F.7 Level 3.2 (Plan Rescue)

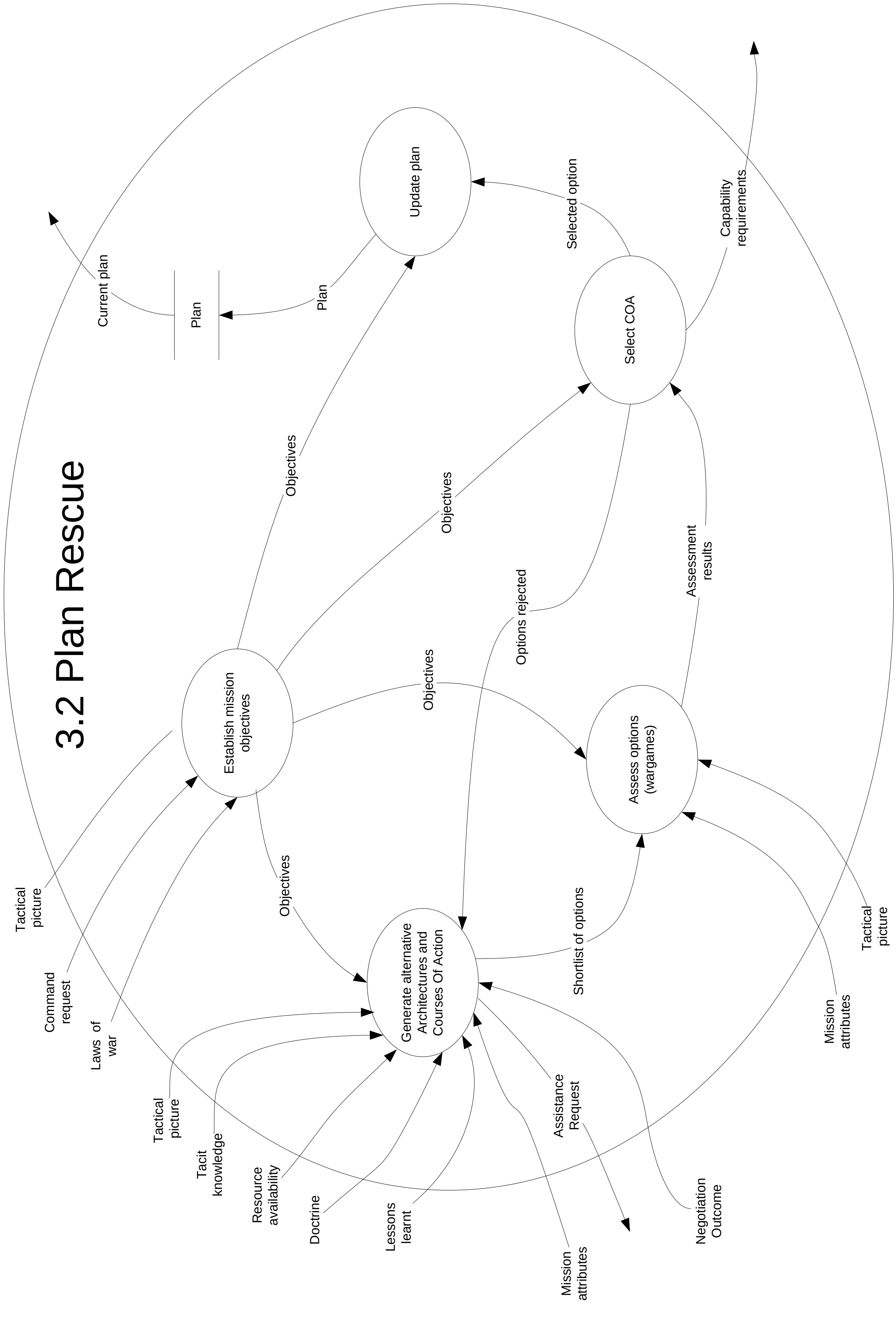


Figure 86 – Level 3.2 (Plan Rescue) FFD

F.8 Level 4 (Execute Action)

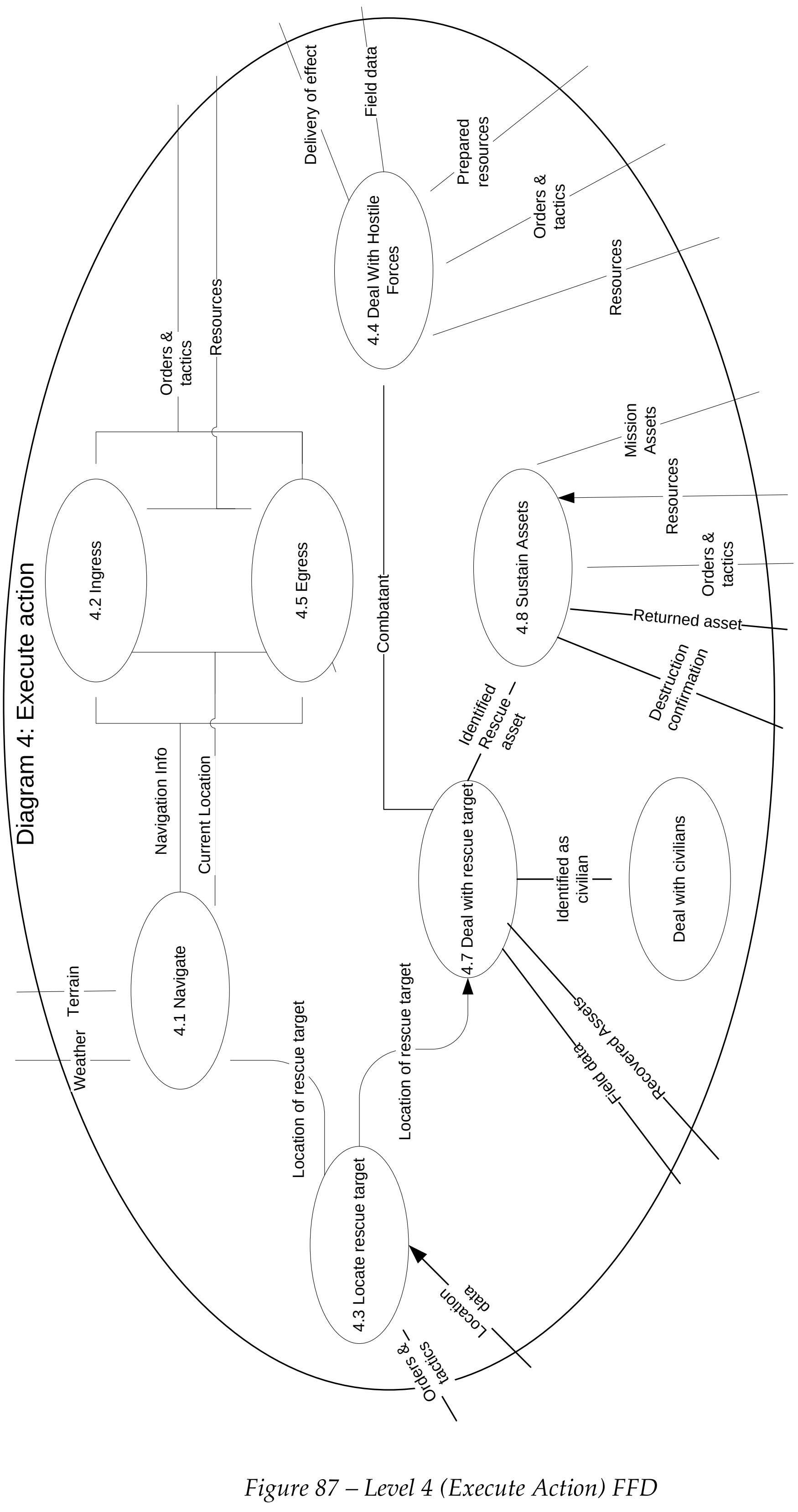


Figure 87 – Level 4 (Execute Action) FFD

F.8.1 Level 4.7 (Deal With Rescued Assets)

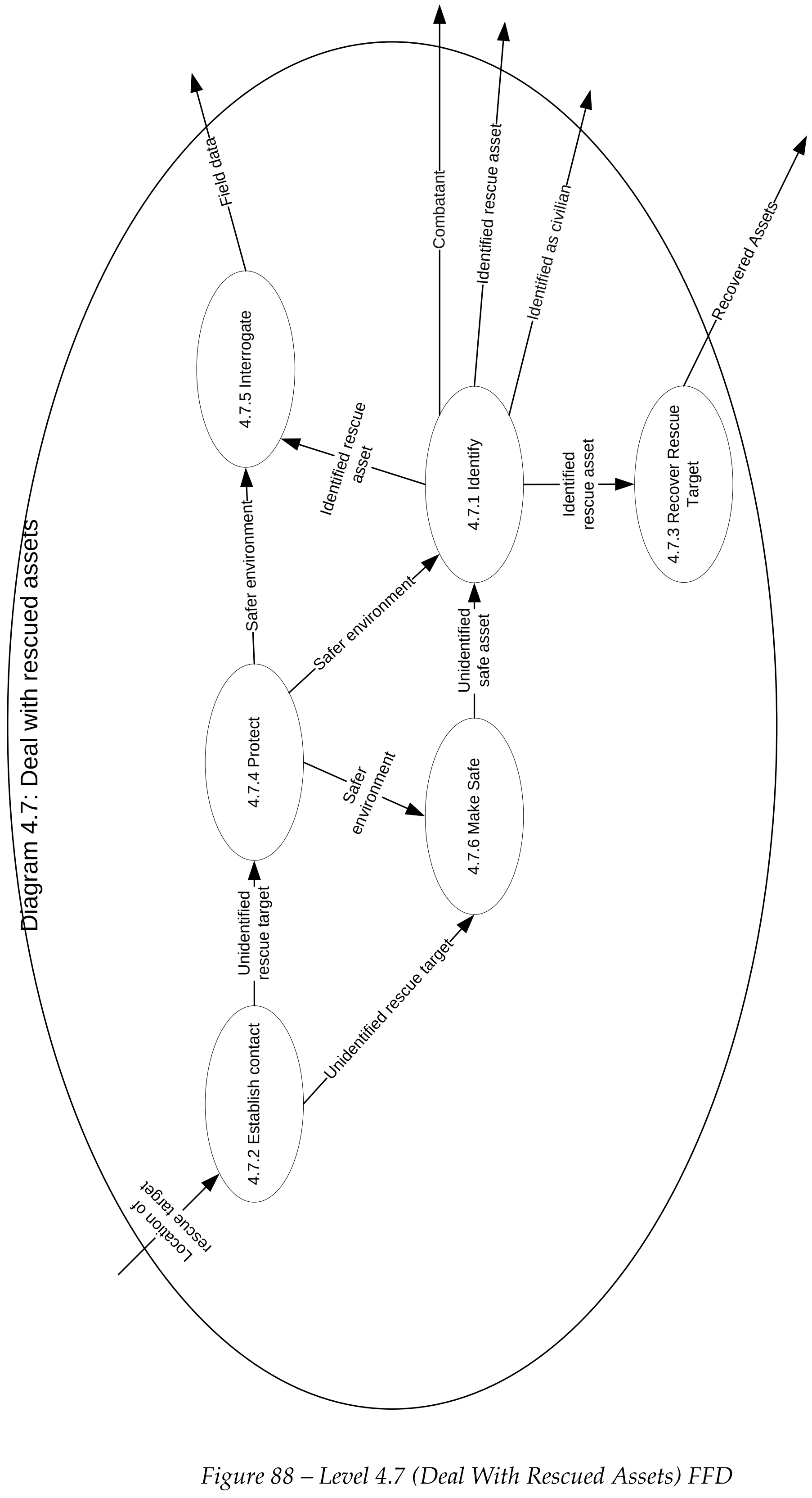


Figure 88 – Level 4.7 (Deal With Rescued Assets) FFD

F.8.2 Level 4.8 (Sustain Assets)

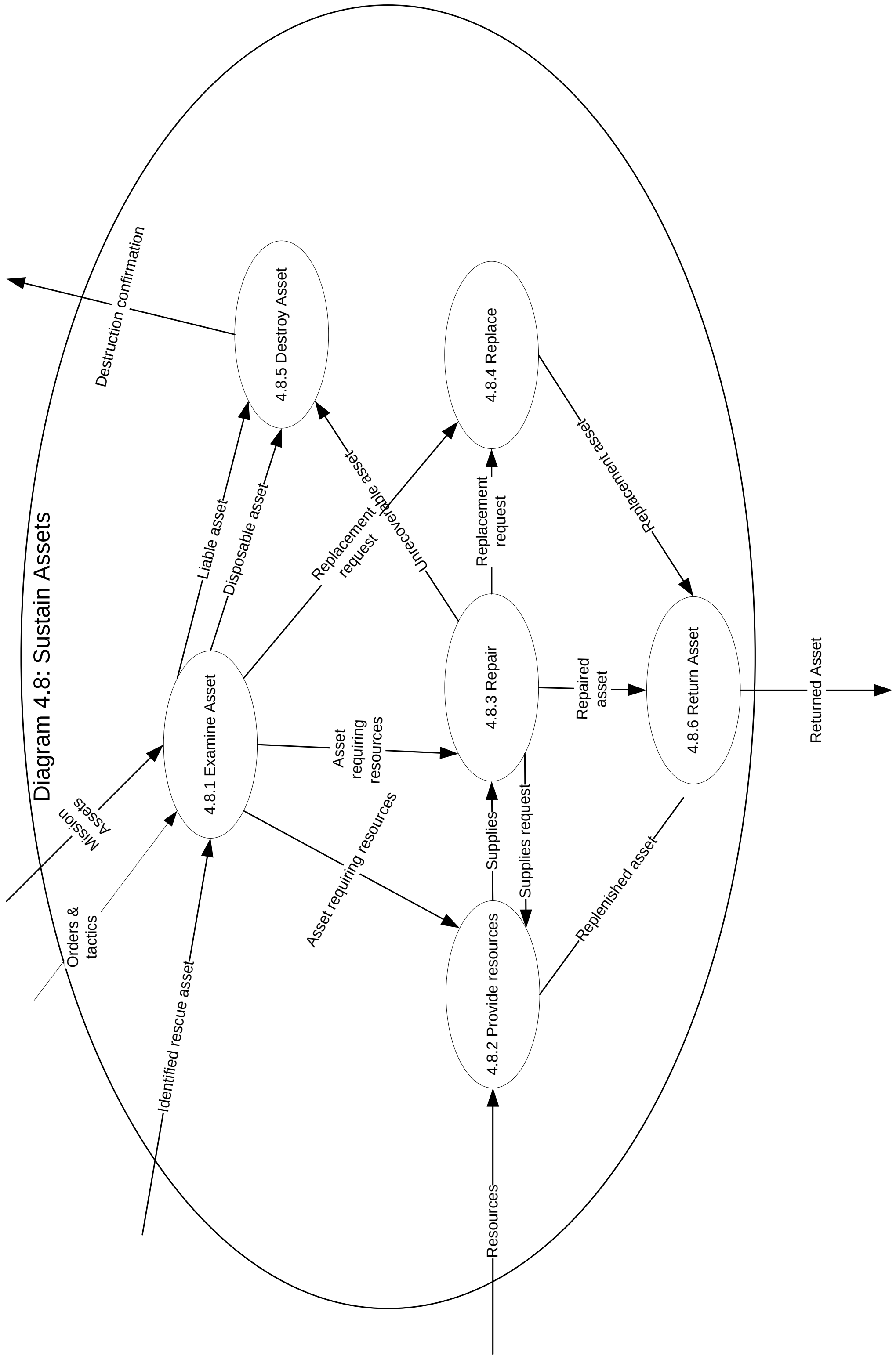


Figure 89 – Level 4.8 (Sustain Assets) FFD

F.9 Level 5 (Command)

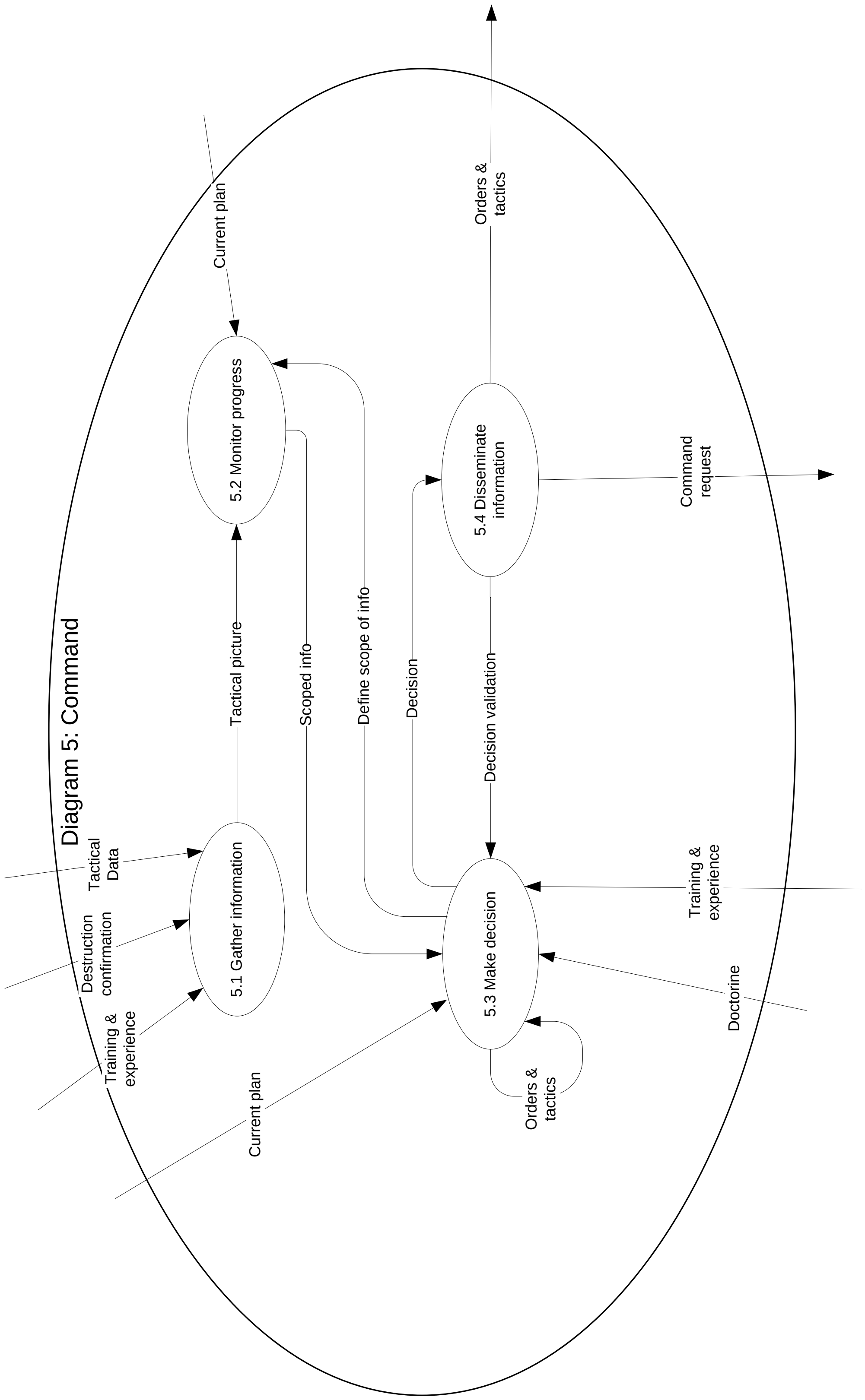


Figure 90 – Level 5 (Command) FFD

F.10 Level 6 (Assess Mission)

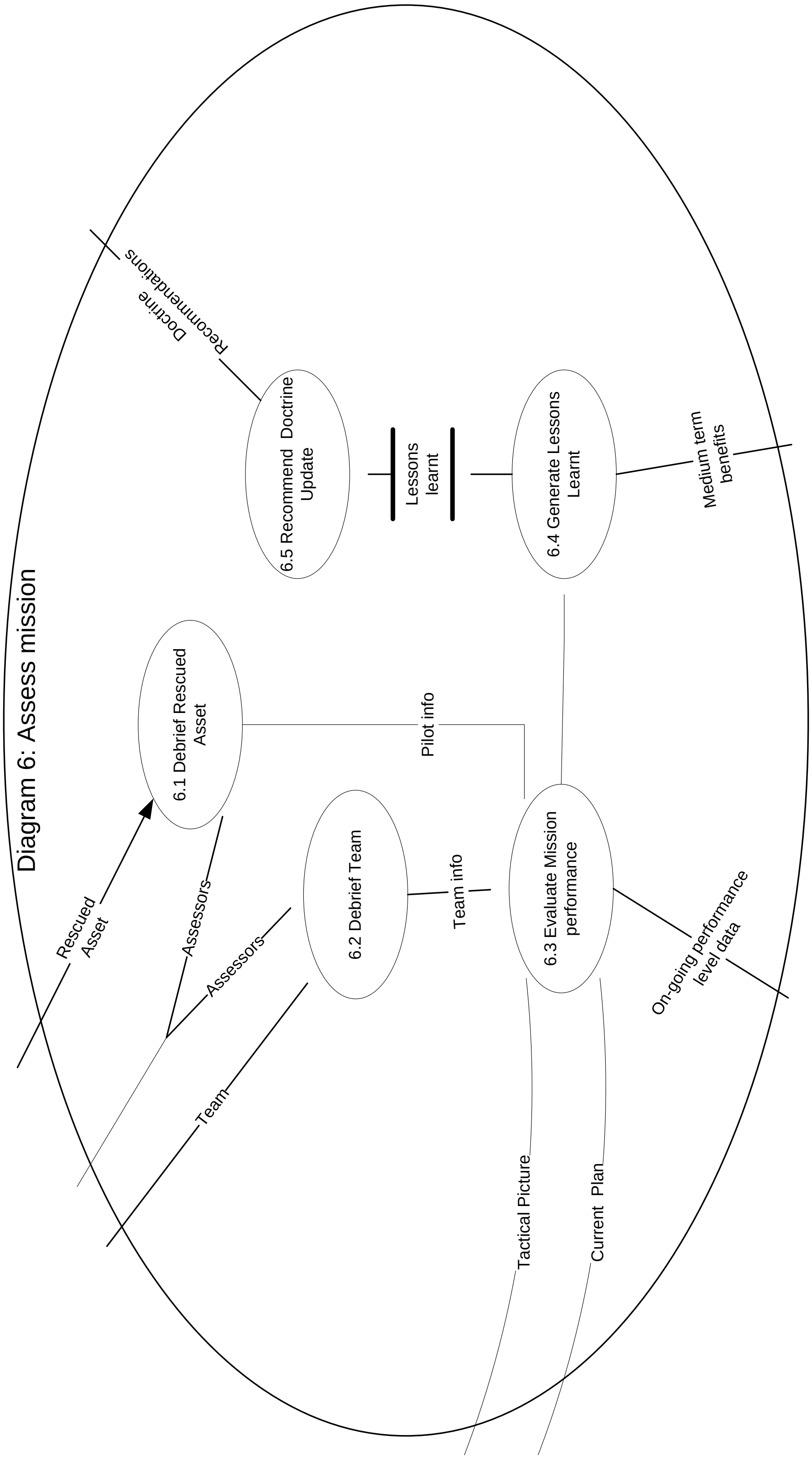


Figure 91 – Level 6 (Assess Mission) FFD

Appendix G

Attributes, Secondary Attributes and Factors

This Appendix presents the full set of Attributes, Secondary Attributes and Factors developed as part of this research.

Table 12 – Full Proposed Set of Attributes, Secondary Attributes and Factors

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
Adaptability	The ability of a system to change how it functions (allocation of functions) and what it does (what functions it executes) in response to the environment.	Mobility	The ability of a system to deliver an effect/output somewhere else (reach)	Footprint	The physical size and weight of the system
				Endurance	The time and distance a system can operate for on its own resources.
				Range	The range at which the system can deliver effects/outputs
Availability	The ability to provide a particular functional level for the majority of time	Maintainability	The ability of a system to be retained in, or restored to, a state in which it can perform its required functions.	Testability	How easily / comprehensively the system can be tested, either by an external party or by self test.
				Supportability	A measure of how easy a system is to support (i.e. the extent, complexity and size of its reach-back architecture)
				Durability	A measure of how long lasting the system is in its normal operating environment.
		Reliability	The degree of certainty in the system to be functionally fit	Redundancy	Duplication of critical functions to enable continuation in case of subsystem

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
					failure.
Cooperability	The ability to engage in co-operative behaviour in a team, e.g. by information sharing and mutual support.	Compatibility	The ability of a system to work harmoniously with others systems under specific conditions to fulfil relevant requirements without causing unacceptable interactions.	Commonality	How much the system has in common with other systems to enable them to work together (protocols, communications, etc)
		Seamlessness	The ability to function consistently and coherently with other systems.		
		Sharability	The extent to which information can be shared between and across a system without loss of fidelity.	Ambiguity	The ambiguity of the information provided by a particular system.
				Sharing	The ability and willingness to share info etc between agents within and across sub systems
		Sociability/ ability to interact	Describes how well the humans in the system get along together and muck in - thinking about team spirit? A def I found is: The relative tendency or disposition to be sociable or associate with one's fellows	Good communicator	The ability to listen and respond appropriately and to communicate meaning in the most appropriate media
				Trustability	Both the ability to trust and be trusted
				Conflict management	The ability to handle and dispel conflict
				Collaborability	The willingness and ability to collaborate and/or cooperate with others
				Transparency	The degree to which an agent is willing/able to be clear about goals / tasks / objective s / boundaries etc

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
				Social orientation	Ability to orient oneself in terms of dealing with other agents depending on cues in the external environment i.e. external to the agent in question
Credibility	The impression created by a system of its intent to follow through on its actions	Acceptability	The satisfactoriness of utilising a system through conformance to approved standards (+ ethical standards?)	Legitimacy	The systems adherence to the relevant laws governing its actions.
				Believability	The wider (public) perception of the system's validity.
				Transparency	The degree to which an agent is willing / able to be clear about goals / tasks / objectives / boundaries etc
				Trustability	Both the ability to trust and be trusted
		Accountability	The ability for a system to be accountable for its actions	Responsibility	A measure of how answerable a system is to someone for something or being responsible for its conduct.
				Transparency	The degree to which an agent is willing / able to be clear about goals / tasks / objectives / boundaries etc
				Trustability	Both the ability to trust and be trusted
Decision Making Superiority	The ability to make the right decision at the right time	Awareness	The system's knowledge of the battlespace	Bandwidth	The bandwidth available to the system for communications.
				Capacity	The amount of information that a system can process
				Newness	Ability to deal with new information
				Predictability	Will follow orders and CONOPS to

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
					produce same or expected results
				Physical orientation	Ability to orient oneself in physical space depending on cues in the external environment i.e. external to the agent in question
				Cognitive orientation	Ability to process external cues to develop a mental picture of the current situation
				Social orientation	Ability to orient oneself in terms of dealing with other agents depending on cues in the external environment i.e. external to the agent in question
		Implications	The ability to apply situational awareness to derive or edit a strategic or tactical overview of a new or on-going mission	Physical orientation	Ability to orient oneself in physical space depending on cues in the external environment i.e. external to the agent in question
				Cognitive orientation	Ability to process external cues to develop a mental picture of the current situation
				Social orientation	Ability to orient oneself in terms of dealing with other agents depending on cues in the external environment i.e. external to the agent in question
				Assess implications	Ability to assess implications of SA and derive a strategic or tactical plan or take account of implications in current mission plan
		Decision Making		Risk tolerance	the ability to manage risk at the appropriate level for the mission

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
				Time variance	Ability to deal with variable time pressures
				Decide	Ability and willingness to make decisions when required and take responsibility for them
				Autonomy	The ability to act autonomously within any boundaries set for permitted and obligatory actions
		Error Management		Retrieval	Ability to recognise when a wrong decision was made or it was based on wrong information, made at wrong time and then can decide appropriate action to retrieve position or remedy error
				Cognitive preparedness	Ability to demonstrate a mindset that expects unpleasant surprises and deals with them
				Flexibility in action	Ability to deploy different modes of adaption in different circumstances as an individual agent
Deployability	A measure of what is required to get "on station" at the required functional level	Mobility	The ability of a system to deliver an effect/output somewhere else (reach)		
		Recoverability	The ability of a system to be undeployed		
Effectiveness	To do something effectively and properly, to deliver the required effect.	Accessibility	The quality of being at hand when needed		
		Configurability	The ability of a system to change its functionality through the rearrangement of its parts.		
		Dependability	A measure of how worthy the system		

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
			is of reliance or trust by other systems.		
		Integrity	The wholeness of the system, everything within the system has a role or purpose and all inputs and outputs have somewhere to go - no internal conflicts		
		Lethality	The ability to kill	Deadliness	The likelihood of mortality occurring from an effect of the system.
		Mobility	The ability of a system to deliver an effect/output somewhere else (reach)		
		Predictability	The ability of a system to consistently produce the same expected results.		
		Timeliness	The ability to functionally achieve something within a predetermined or favourable timeframe.		
		Usability	How usable the system is by the humans within it (who are likely to have been trained/specialists) including recovery from errors. This does not imply simplicity.		
		Processing capability	The ability to process incoming knowledge, information / data effectively, efficiently and in a timely manner, to reach conclusions regardless of the state of the information / data	Speed	The ability to process information rapidly
				Ambiguousness	The ability to deal with ambiguous info etc
				Complexity	The ability to deal

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
					with complex info etc
				Contradictions	The ability to deal with contradictory info etc
				Uncertainty (Value)	The ability to deal with info etc with is not clear in terms of its content, meaning or source
				Incompleteness (Quality)	The ability to deal with info etc with is not complete in terms of its content or meaning
				Objectivity	The ability to objectively analyse technical data
				Prioritisation	The ability or possession of sufficient expertise to prioritise incoming info etc
				Sharing	The ability and willingness to share info etc between agents within and across sub systems
Flexibility	The allocation of functions within a system	Configurability	The ability of a system to change its functionality through the rearrangement of its parts.		
		Interchangeability	The ability to allocate functions to multiple subsystems without needing to alter the subsystems themselves.		
Interoperability	The ability to operate in synergy in the execution of assigned tasks.	Compatibility	The ability of a system to work harmoniously with others systems under specific conditions to fulfil relevant requirements without causing unacceptable interactions.	Commonality	How much the system has in common with other systems to enable them to work together (protocols, communications, etc)
		Sharability	The extent to which information	Ambiguity	The ambiguity of the information

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
			can be shared between and across a system without loss of fidelity.		provided by a particular system.
Orientability	The ability to comprehend the environment				
Survivability	The ability to function during and after a natural or man-made disturbance	Resilience	The degree to which a system can recover from a perturbation in its environment.	Stability	A measure of how much variation a system experiences in normal operating conditions.
				Durability	A measure of how long lasting the system is in its normal operating environment.
		Robustness	The degree of tolerance that a system has to continue functioning when subjected to perturbations in its environment.		
		Security	A measure of a systems protection against the unauthorized use of and prevent unauthorized access to the systems and it's subsystems.		
		Susceptibility	The extent to which a system is likely to be found, targeted and hit.		
		Vulnerability	The consequences of being hit		
Sustainability	The ability to deliver a level of performance despite any interference	Degradability	The degree of functional impairment a system will suffer when exposed to the environment for prolonged periods		
		Dependability	A measure of how worthy the system is of reliance or trust by other systems.	Trustworthiness	How much a system can be trusted by another system.
		Resilience	The degree to which a system can recover from a	Stability	A measure of how much variation a system

Attribute	Attribute Definition	Secondary Attribute	Secondary Attribute Definition	Factor	Factor Definition
			perturbation in its environment.		experiences in normal operating conditions.
				Durability	A measure of how long lasting the system is in its normal operating environment.
		Robustness	The degree of tolerance that a system has to continue functioning when subjected to perturbations in its environment.		

Appendix H

Physical Environment Characteristics

This Appendix presents the full set of Physical Environment Characteristics developed as part of this research.

Table 13 – Physical Environment Characteristics

ID	Group	ID	Subgroup	*Description*	ID	Element	*Description*	ID(4)	*Description*
1	Climate (Köppen climate classification)	1.1	GROUP A: Tropical/megathermal climates	*Tropical climates with abundant precipitation during a portion of the year. Mean monthly climates must exceed 18ºC; precipitation exceeds evaporation.*	1.1.2	Tropical rain forest climate			
					1.1.3	Tropical monsoon climate			
					1.1.4	Tropical wet and dry or savannah climate			
		1.2	GROUP B: Dry (arid and semiarid) climates	*Arid and semiarid climates of the low and middle latitudes. Evaporation exceeds precipitation.*					
		1.3	GROUP C: Temperate / mesothermal climates	*Mild and humid climates primarily in the lower middle latitudes. Mean monthly temperature of the coldest month must be between 18 and –3ºC; at least one month must have a mean temperature of 10 ºC or higher.*	1.3.1	Mediterranean climates			
					1.3.2	Humid subtropical climates			
					1.3.3	Maritime Temperate climates or Oceanic climates			
					1.3.4	Maritime Subarctic climates or Sub polar Oceanic climates			
		1.4	GROUP D: Continental/micro thermal climate	*Found in the upper middle latitudes and sub polar regions of the northern hemisphere.*	1.4.1	Hot Summer Continental climates			

ID	Group	ID	Subgroup	*Description*	ID	Element	*Description*	ID(4)	*Description*
				Humid continental climates with cold winters; mean monthly temperature of coldest month below -3ºC; mean monthly temperature of the warmest month must be 10ºC or higher.					
					1.4.2	Warm Summer Continental or Hemiboreal climates			
					1.4.3	Continental Subarctic or Boreal (taiga) climates			
					1.4.4	Continental Subarctic climates with extremely severe winters			
		1.5	GROUP E: Polar climates	*Cold climates of the northern latitudes. All months must average below 10ºC.*	1.5.1	Tundra climate			
					1.5.2	Ice Cap climate			
2	Environment	2.1	Land	*Mission environment is on land*	2.1.1	Relief	*A profile of the relief of the terrain in the environment*	2.1.1.1	Mountainous
								2.1.1.2	Hills / Valleys
								2.1.1.3	Plains
					2.1.2	Vegetation	*A profile of the vegetation in the environment*	2.1.2.1	Tundra
								2.1.2.2	Plains
								2.1.2.3	Desert
								2.1.2.4	Forest
								2.1.2.5	Jungle
		2.2	Sea	*Mission environment is at sea*	2.2.1	Sea Conditions	*The conditions at sea*	2.2.1.1	Beaufort wind

ID	Group	ID	Subgroup	*Description*	ID	Element	*Description*	ID(4)	*Description*
							whilst the mission is conducted		force scale (1-12)
								2.2.1.2	Sea Temperature (degrees centigrade)
		2.3	Littoral	*Mission environment is based around the coast - consider both Land and Sea cases above*					
3	Predictability	3.1	Predictable	*Situation is predictable, few if any variables present*					
		3.2	Unpredictable	*Situation is unpredictable due to variables present in the battlespace*					
		3.3	Static	*Situation is considered to be static - unlikely to change in the near future (whilst the mission is conducted)*					
		3.4	Changing	*Situation is considered to be changing, sources of volatility present in the environment*					
4	Strategic Purpose	4.1	Offensive	*Dominance over the opposition is the primarily strategic purpose*					
		4.2	Defensive	*Resistance to the opposition is the primary strategic purpose*					
		4.3	OOTW	*Strategic purpose is an OOTW (Operation Other Than War)*					
5	Political Environment	5.1	Local Hostility	*A measure of the local hostility to friendly systems*	10.1.1	Friendly	*Locals will provide help and assistance*		
					10.1.2	Benign	*Locals will provide help and assistance on request*		
					10.1.3	Low	*Locals may provide help and assistance on request; locals may pose a threat to friendly assets*		

ID	Group	ID	Subgroup	*Description*	ID	Element	*Description*	ID(4)	*Description*
					10.1.4	Medium	*Locals unlikely to provide help and assistance; locals likely to pose a threat to friendly assets*		
					10.1.5	High	*Locals highly unlikely to provide help and assistance; locals highly likely to pose a threat to friendly assets*		
		5.2	Political Stability	*A measure of the stability of the environment's political system*	10.2.1	Despotism	*Locality controlled by a single authoritarian body*		
					10.2.2	High	*Locality controlled by a recognised political power whose control is unchallenged*		
					10.2.3	Medium	*Locality controlled by a recognised political power whose control is unlikely to change*		
					10.2.4	Low	*Locality controlled by a political power whose control is likely to change*		
					10.2.5	Anarchy	*Locality not controlled by any political power, either controlled by multiple competing powers or none at all*		

Appendix I
Case Study One – Seawolf CSAR in Vietnam

I.1 Sequence of Events

The first aeromedical helicopter evacuation unit in Vietnam was the 57th Medical Detachment from Fort Meade, Maryland. The name "Dustoff" was derived from the radio call-sign given to this unit, which flew 5 Huey helicopters and was service to the then 8,000 troops in Vietnam. For the pilots flying the mission their only company was "Paddy Control", the Air Force controllers who provided RADAR coverage for most of the Delta.

Table 14 – Case Study One Sequence of Events

1	Paddy Control contacts Dustoff Operations with an emergency scramble mission - a Navy Seawolf helicopter had given a mayday call near Tra Vinh indicating that it had been hit by enemy fire and was going down.
2	Captain David Freeman, a Dustoff pilot from the 57th Medical Detachment, and his crew scramble.
3	Paddy vectors Freeman to the coordinates where they had last seen the downed Seawolf on RADAR.
4	Freeman takes approximately 15 minutes to get there.
5	Freeman starts a search pattern which involves flying north to south with the crew visually searching for the crashed helicopter or a signal from their crew. At this stage Freeman does not know if anyone has survived or the tactical situation.
6	Within a couple of minutes the crew chief spots and points out the wreckage of the helicopter. An initial assessment of the situation by Freeman and his crew is:
6.1	The downed Seawolf is in an open area, easily visible and with plenty of room next to it for the Huey to land.
6.2	The downed Seawolf looks in good physical condition with little external damage.
6.3	From a tactical perspective the location is bad as the open area is difficult to defend with little cover.
7	Freeman makes a low pass over the helicopter and sees four airmen. They are huddled next to the downed Seawolf and holding off a squad of Vietcong with their personal weapons. The downed Seawolf has a 50-calibre machine gun mounted on it but it was facing the wrong way and could not be swung around for defence.
8	Freeman gets a call on UHF radio from another Seawolf helicopter which had responded to the emergency call. The pilot told them he was five minutes away and would give them gun cover whilst they made the pickup.
9	Whilst waiting for the Seawolf to arrive the Dustoff crew decide on what appeared to be the best plan of action.
10	Freeman approaches the crash site facing the enemy location and, using the downed helicopter as cover, land within 20-25 yards of where the downed crew were making their stand. As the downed crew ran towards the rescue helicopter the Seawolf arrived and fired on the enemy position with machine gun fire and rockets. This caused the Viet Cong to cease-fire and dig in, allowing the downed crew to scramble onto the Dustoff and Freeman exited the way that they came in without taking any hits.

I.2 Functional Profile

The following section presents the functional profiles generated for each of the available assets.

I.2.1 Functional Profile: Huey Helicopter (Rescue Configuration)

Table 15 – Functional Profile Huey Helicopter (Rescue Configuration)

	Name	**Huey Helicopter (Rescue Configuration)**		
	Basic Configuration	Huey Helicopter, Pilot, Co-Pilot/Navigator, Crew Chief, Medic		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	Y
		1.2	Collate distress information	N
		1.3	Validate distress information	N
		1.4	Acknowledge receipt	Y
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	Y
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	Y
		4.2	Ingress	Y
		4.3	Locate rescue target	Y
		4.4	Deal with hostile forces	N
		4.5	Egress	Y
		4.6	Deal with civilians	Y
		4.7	Deal with rescue target	Y
		4.8	Sustain assets	Y
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

I.2.2 Functional Profile: Huey Helicopter (Attack Configuration)

Table 16 – Functional Profile Huey Helicopter (Attack Configuration)

	Name	**Huey Helicopter (Attack Configuration)**		
	Basic Configuration	Huey Helicopter, Pilot, Co-Pilot/Navigator		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	Y
		1.2	Collate distress information	N
		1.3	Validate distress information	N
		1.4	Acknowledge receipt	Y
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	Y
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	Y
		4.2	Ingress	Y
		4.3	Locate rescue target	Y
		4.4	Deal with hostile forces	Y
		4.5	Egress	Y
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

I.2.3 Functional Profile: Air Force Controllers

Table 17 – Functional Profile Air Force Controllers

	Name	**Air Force Controllers - "Paddy Control"**		
	Basic Configuration	Air Force Controllers, radar infrastructure, communications infrastructure		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	Y
		1.2	Collate distress information	Y
		1.3	Validate distress information	Y
		1.4	Acknowledge receipt	Y
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	N
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	N
		4.2	Ingress	N
		4.3	Locate rescue target	N
		4.4	Deal with hostile forces	N
		4.5	Egress	N
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	N
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

I.2.4 Functional Profile: 57th Medical Attachment Controllers

Table 18 – 57th Medical Attachment Controllers

	Name	**57th Medical Attachment Controllers - "Dustoff Operations"**		
	Basic Configuration	Air Force Controllers, RADAR infrastructure, communications infrastructure		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	N
		1.2	Collate distress information	N
		1.3	Validate distress information	N
		1.4	Acknowledge receipt	N
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	Y
		3.3	Select appropriate resources	Y
		3.4	Prepare resources	Y
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	N
		4.2	Ingress	N
		4.3	Locate rescue target	N
		4.4	Deal with hostile forces	N
		4.5	Egress	N
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

I.2.5 Functional Profile – Forming Systems

Matrix 24 – Case Study One Functional Profile

	Name		Name	Mission	(1) Rescue Huey	(2) Attack Huey	(3) Paddy Control	(4) Dustoff Ops	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Detect distress call	1.1	Receive distress call	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.2	Collate distress information	Y	N	N	Y	N	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
		1.3	Validate distress information	Y	N	N	Y	N	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
		1.4	Acknowledge receipt	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
2	Collaborate with coalition	2.1	Vet information for security	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		2.2	Transfer knowledge	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		2.3	Negotiate assistance	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
3	Planning & Adapting	3.1	Assess situation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.2	Plan rescue	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.3	Select appropriate resources	Y	N	N	N	Y	FAIL	FAIL	FAIL	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.4	Prepare resources	Y	N	N	N	Y	FAIL	FAIL	FAIL	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.5	Conform to the laws of war	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
4	Execute Action	4.1	Navigate	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	FAIL
		4.2	Ingress	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	FAIL
		4.3	Locate rescue target	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	FAIL
		4.4	Deal with hostile forces	Y	N	Y	N	N	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	Y	FAIL

	Name		Name	Mission	*(1) Rescue Huey*	*(2) Attack Huey*	*(3) Paddy Control*	*(4) Dustoff Ops*	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
		4.5	Egress	Y	Y	Y	N	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	FAIL
		4.6	Deal with civilians	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
		4.7	Deal with rescue target	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
		4.8	Sustain assets	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
5	Command	5.1	Gather information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.2	Monitor progress	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.3	Make decision	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.4	Disseminate information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
6	Assess Mission	6.1	Debrief rescued asset	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.2	Debrief team	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.3	Evaluate mission performance	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.4	Generate lessons learnt	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.5	Recommend doctrine update	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N

I.3 Non-Functional Profile

Table 19 – Non-Functional Profile

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Adaptability	1.1	Mobility	1.1.1	Footprint	Small / Medium / Large	Small	small	small	small	small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small
				1.1.2	Endurance	Short / Medium / Long	Short	Short	Short	Long	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL
				1.1.3	Range	Short / Medium / Long	Medium	Medium	Medium	Long	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL
2	Availability	2.1	Maintainability	2.1.1	Testability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				2.1.2	Supportability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				2.1.3	Durability	Short / Medium / Long	Medium	Medium	Medium	Long	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL
		2.2	Reliability	2.2.1	Redundancy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
3	Cooperability	3.1	Compatibility	3.1.1	Commonality	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.2	Seamlessness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.3	Sharability	3.3.1	Ambiguity	High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				3.3.2	Sharing	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.4	Sociability/ ability to interact	3.4.1	Good communicator	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.2	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.3	Conflict management	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.4	Collaborability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.5	Transparency	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.6	Social orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
4	Credibility	4.1	Acceptability	4.1.1	Legitimacy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.1.2	Believability	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				4.1.3	Transparency	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.1.4	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
		4.2	Accountability	4.2.1	Responsibility	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.2.2	Transparency	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				4.2.3	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
5	Decision Making Superiority	5.1	Awareness	5.1.1	Bandwidth	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.2	Capacity	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.3	Newness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.1.4	Predictability	High / Low	n/a	Low	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.5	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.1.6	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.1.7	Social Orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		5.2	Situational Awareness Synthesis	5.2.1	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				5.2.2	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.2.3	Social orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.4	Assess implications	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		5.3	Decision Making	5.3.1	Risk Tolerance	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.2	Time variance	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.3	Decide	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.4	Autonomy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		5.4	Error Management	5.4.1	Retrieval	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.4.2	Cognitive preparedness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.4..3.	Flexibility in action	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
6	Deployability	6.1	Mobility	6.1.1	Footprint	Small / Medium / Large	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				6.1.2	Endurance	Short / Medium / Long	Short	Short	Short	Long	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL
				6.1.3	Range	Short / Medium / Long	Medium	Medium	Medium	Long	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL
		6.2	Recoverability			High / Low	High	High	High	Low	Low	High	High	High	High	High	High	High	High	High	High	FAIL
7	Effectiveness	7.1	Accessibility			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.2	Configurability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		7.3	Dependability			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.4	Integrity			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.5	Lethality	7.5.1	Deadliness	High / Low	High	Low	High	Low	Low	High	High	FAIL	High	High	High	FAIL	High	FAIL	High	FAIL
		7.6	Mobility	6.1.1	Footprint	Small / Medium / Large	Small	small	small	small	small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small
				6.1.2	Endurance	Short / Medium / Long	Short	Short	Short	Long	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL
				6.1.3	Range	Short / Medium / Long	Medium	Medium	Medium	Long	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
		7.7	Awareness	7.7.1	Bandwidth	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.2	Capacity	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.3	Newness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.4	Predictability	High / Low	n/a	Low	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.5	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.6	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.7	Social Orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		7.8	Timeliness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.9	Usability			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.10	Processing Capability	7.10.1	Speed	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.2	Ambiguousness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				7.10.3	Complexity	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.10.4	Contradictions	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.5	Uncertainty (Value)	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.6	Incompleteness (Quality)	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.7	Objectivity	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.8	Prioritisation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.9	Sharing	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
8	Flexibility	8.1	Configurability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		8.2	Interchangeability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
9	Interoperability	9.1	Compatibility	9.1.1	Commonality	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		9.2	Sharability	9.2.1	Ambiguity	High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
10	Orientability					High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
11	Survivability	11.1	Resilience	11.1.1	Stability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				11.1.2	Durability	Short / Medium / Long	Medium	Medium	Medium	Long	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL
		11.2	Robustness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		11.3	Security			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		11.4	Susceptibility			High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
		11.5	Vulnerability			High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
12	Sustainability	12.1	Degradability			High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
		12.2	Dependability	12.2.1	Trustworthiness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		12.3	Resilience	12.3.1	Stability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				12.3.2	Durability	High / Low	Long	Medium	Medium	Long	Long	FAIL	Long	Long	Long	Long	Long	Long	Long	Long	Long	Long

ID(1)	Attribute	ID(2)	Secondary Attribute	ID(3)	Factor	*Measures*	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
		12.4	Robustness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

I.4 Physical Environment Profile

Matrix 25 – Case Study One Physical Environment Profile

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Climate (Köppen climate classification)	1.1	GROUP A: Tropical/megathermal climates	1.1.2	Tropical rain forest climate	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.1.3	Tropical monsoon climate	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.1.4	Tropical wet and dry or savannah climate	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.2	GROUP B: Dry (arid and semiarid) climates			n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		1.3	GROUP C: Temperate/mesothermal climates	1.3.1	Mediterranean climates	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.3.2	Humid subtropical climates	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.3.3	Maritime Temperate climates or	n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
					Oceanic climates																
				1.3.4	Maritime Subarctic climates or Sub polar Oceanic climates	n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		1.4	GROUP D: Continental/micro thermal climate	1.4.1	Hot Summer Continental climates	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.4.2	Warm Summer Continental or Hemiboreal climates	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.4.3	Continental Subarctic or Boreal (taiga) climates	n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.4.4	Continental Subarctic climates	n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
					with extremely severe winters																
		1.5	GROUP E: Polar climates	1.5.1	Tundra climate	n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.5.2	Ice Cap climate	n/a	N	N	N	N	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
2	Environment	2.1	Land	2.1.1	Relief	n/a	N	N	N	N	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
						Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				2.1.2	Vegetation	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
						n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		2.2	Sea	2.2.1	Sea Conditions	n/a	4-5?	4-5?			FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						n/a	ALL	ALL			FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		2.3	Littoral			n/a	Y	Y			FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
3	Predictability	3.1	Predictable			n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		3.2	Unpredictable			Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.3	Static			n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		3.4	Changing			Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
4	Strategic Purpose	4.1	Offensive			Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.2	Defensive			n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		4.3	OOTW			n/a	Y	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
5	Political Environment	5.1	Local Hostility	5.1.1	Friendly	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.2	Benign	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.3	Low	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	Mission	Rescue Huey	Attack Huey	Paddy Control	Dustoff Operations	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				5.1.4	Medium	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				5.1.5	High	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		5.2	Political Stability	5.2.1	Despotism	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				5.2.2	High	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.3	Medium	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.4	Low	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.5	Anarchy	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

Appendix J

Case Study Two – Rescue Of Lt Devon Jones

J.1 Sequence of Events

Table 20 – Case Study Two Sequence of Events

1	AWACS receives mayday call from Slade as he parachutes down after ejecting from his aircraft.
2	Jones and Slade land some distance away from each other
3	Jones digs in and camouflages himself
4	MH-53J 'Pave Low' helicopter scrambled to crash site
5	Iraqi MiG-23 fighters appear, engaged by two USAF F-15Cs on RESCAP (Combat Air Patrol), MiG-23's retreat
6	At approximately 10:30 Jones hides from what he thinks are enemy fighters, actually the two F-15Cs on RESCAP
7	At approximately 10:30 Slade is discovered and detained by Iraqi forces
8	MH-53J 'Pave Low' helicopter returns to Arar to refuel, returning immediately to continue search
9	'Sandy' A-10A makes contact with Jones via radio on SAR frequency
10	Jones guides the 'Sandy' A-10A to him, 'Sandy' A-10A locates him exactly using its INS
11	Iraqi MiG-23s move into area, engaged by two USAF F-15Cs on RESCAP (Combat Air Patrol), MiG-23's retreat
12	'Sandy' A-10 sanitises the area
13	MH-53J 'Pave Low' moves in to rescue Jones, spots an Iraqi truck en route to Jones' position
14	'Sandy' A-10A engages Iraqi truck
15	MH-53J 'Pave Low' lands near destroyed Iraqi truck, Jones breaks cover and boards the helicopter, helicopter returns to base
16	'Sandy A-10A's stay on station to assure the MH-53J's safety

J.2 Functional Profile

The following section presents the functional profiles generated for each of the available assets.

J.2.1 Functional Profile: MH-53J 'Pave Low' Helicopter

Table 21 – Functional Profile: MH-53J 'Pave Low' Helicopter

	Name	**MH-53J 'Pave Low' Helicopter**		
	Basic Configuration	Huey Helicopter, Pilot, Co-Pilot/Navigator, Crew Chief, Medic		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	N
		1.2	Collate distress information	N
		1.3	Validate distress information	N
		1.4	Acknowledge receipt	N
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	Y
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	Y
		4.2	Ingress	Y
		4.3	Locate rescue target	Y
		4.4	Deal with hostile forces	N
		4.5	Egress	Y
		4.6	Deal with civilians	Y
		4.7	Deal with rescue target	Y
		4.8	Sustain assets	Y
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

J.2.2 Functional Profile: F-15C 'Eagle' Tactical Fighter Aircraft

Table 22 – Functional Profile: F-15C 'Eagle' Tactical Fighter Aircraft

	Name	**F-15C 'Eagle' Tactical Fighter Aircraft**		
	Basic Configuration	Pilot, assorted weaponry (One internally mounted M-61A1 20mm 20-mm, six-barrel cannon with 940 rounds of ammunition; four AIM-9L/M Sidewinder and four AIM-7F/M Sparrow air-to-air missiles, or eight AIM-120 AMRAAMs, carried externally.)		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	N
		1.2	Collate distress information	N
		1.3	Validate distress information	N
		1.4	Acknowledge receipt	N
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	N
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	Y
		4.2	Ingress	Y
		4.3	Locate rescue target	Y
		4.4	Deal with hostile forces	Y
		4.5	Egress	Y
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	N
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

J.2.3 Functional Profile: Sandy' A-10A Close Air Support Aircraft

Table 23 – Functional Profile: Sandy' A-10A Close Air Support Aircraft

	Name	**Sandy' A-10A Close Air Support Aircraft**		
	Basic Configuration	Pilot, assorted weaponry (One 30 mm GAU-8/A seven-barrel Gatling gun; up to 16,000 pounds (7,200 kilograms) of mixed ordnance on eight under-wing and three under-fuselage pylon stations, including 500 pound (225 kilograms) Mk-82 and 2,000 pounds (900 kilograms) Mk-84 series low/high drag bombs, incendiary cluster bombs, combined effects munitions, mine dispensing munitions, AGM-65 Maverick missiles and laser-guided/electro-optically guided bombs; infrared countermeasure flares; electronic countermeasure chaff; jammer pods; 2.75-inch (6.99 centimetres) rockets; illumination flares and AIM-9 Sidewinder missiles.)		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	Y
		1.2	Collate distress information	Y
		1.3	Validate distress information	Y
		1.4	Acknowledge receipt	Y
2	Collaborate with coalition	2.1	Vet information for security	N
		2.2	Transfer knowledge	N
		2.3	Negotiate assistance	N
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	N
		3.3	Select appropriate resources	N
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	Y
		4.2	Ingress	Y
		4.3	Locate rescue target	Y
		4.4	Deal with hostile forces	Y
		4.5	Egress	Y
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

J.2.4 Functional Profile: USAF Rescue Co-ordination Centre

Table 24 – Functional Profile: USAF Rescue Co-ordination Centre

	Name	**USAF Rescue Co-ordination Centre (RCC)**		
	Basic Configuration	Various controllers, commanders etc		
				Subsystem Rating
1	Detect distress call	1.1	Receive distress call	N
		1.2	Collate distress information	Y
		1.3	Validate distress information	Y
		1.4	Acknowledge receipt	N
2	Collaborate with coalition	2.1	Vet information for security	Y
		2.2	Transfer knowledge	Y
		2.3	Negotiate assistance	Y
3	Planning & Adapting	3.1	Assess situation	Y
		3.2	Plan rescue	Y
		3.3	Select appropriate resources	Y
		3.4	Prepare resources	N
		3.5	Conform to the laws of war	Y
4	Execute Action	4.1	Navigate	N
		4.2	Ingress	N
		4.3	Locate rescue target	N
		4.4	Deal with hostile forces	N
		4.5	Egress	N
		4.6	Deal with civilians	N
		4.7	Deal with rescue target	N
		4.8	Sustain assets	N
5	Command	5.1	Gather information	Y
		5.2	Monitor progress	Y
		5.3	Make decision	Y
		5.4	Disseminate information	Y
6	Assess Mission	6.1	Debrief rescued asset	N
		6.2	Debrief team	N
		6.3	Evaluate mission performance	N
		6.4	Generate lessons learnt	N
		6.5	Recommend doctrine update	N

J.2.5 Functional Profile

Table 25 – Functional Profile

ID(1)	Name	ID(2)	Name	Mission	MH-53J 'Pave Low'	F-15C 'Eagle'	Sandy' A-10A	USAF RCC	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Detect distress call	1.1	Receive distress call	Y	N	N	Y	N	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
		1.2	Collate distress information	Y	N	N	Y	Y	FAIL	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.3	Validate distress information	Y	N	N	Y	Y	FAIL	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.4	Acknowledge receipt	Y	N	N	Y	N	FAIL	Y	Y	Y	Y	FAIL	Y	Y	FAIL	FAIL	Y
2	Collaborate with coalition	2.1	Vet information for security	N	N	N	N	Y	N	N	N	N	N	N	N	N	N	N	N
		2.2	Transfer knowledge	N	N	N	N	Y	N	N	N	N	N	N	N	N	N	N	N
		2.3	Negotiate assistance	N	N	N	N	Y	N	N	N	N	N	N	N	N	N	N	N
3	Planning & Adapting	3.1	Assess situation	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.2	Plan rescue	Y	Y	N	N	Y	Y	Y	Y	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.3	Select appropriate resources	Y	N	N	N	Y	FAIL	FAIL	FAIL	FAIL	Y	Y	Y	Y	Y	Y	Y
		3.4	Prepare resources	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		3.5	Conform to the laws of war	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
4	Execute Action	4.1	Navigate	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.2	Ingress	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.3	Locate rescue target	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.4	Deal with hostile forces	Y	N	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	FAIL	Y	Y
		4.5	Egress	Y	Y	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.6	Deal with civilians	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL

ID(1)	Name	ID(2)	Name	Mission	*MH-53J 'Pave Low'*	*F-15C 'Eagle'*	*Sandy' A-10A*	*USAF RCC*	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
		4.7	Deal with rescue target	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
		4.8	Sustain assets	Y	Y	N	N	N	Y	Y	Y	FAIL	Y	Y	Y	FAIL	Y	FAIL	FAIL
5	Command	5.1	Gather information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.2	Monitor progress	Y	Y	N	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.3	Make decision	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		5.4	Disseminate information	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
6	Assess Mission	6.1	Debrief rescued asset	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.2	Debrief team	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.3	Evaluate mission performance	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.4	Generate lessons learnt	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N
		6.5	Recommend doctrine update	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N	N

J.3 Non-Functional Profile

Table 26 – Non-Functional Profile

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Adaptability	1.1	Mobility	1.1.1	Footprint	Small / Medium / Large	Small	small	small	small	small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small
				1.1.2	Endurance	Short / Medium / Long	Short	Short	Medium	Short	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL	Short
				1.1.3	Range	Short / Medium / Long	Medium	Medium	Long	Medium	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL	Medium
2	Availability	2.1	Maintainability	2.1.1	Testability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				2.1.2	Supportability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				2.1.3	Durability	Short / Medium / Long	Medium	Medium	Medium	Medium	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium
		2.2	Reliability	2.2.1	Redundancy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
3	Cooperability	3.1	Compatibility	3.1.1	Commonality	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.2	Seamlessness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.3	Sharability	3.3.1	Ambiguity	High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
				3.3.2	Sharing	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		3.4	Sociability/ ability to interact	3.4.1	Good communicator	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.2	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				3.4.3	Conflict management	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.4	Collaborability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.5	Transparency	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				3.4.6	Social orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
4	Credibility	4.1	Acceptability	4.1.1	Legitimacy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.1.2	Believability	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				4.1.3	Transparency	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.1.4	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		4.2	Accountability	4.2.1	Responsibility	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				4.2.2	Transparency	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				4.2.3	Trustability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
5	Decision Making Superiority	5.1	Awareness	5.1.1	Bandwidth	High / Low	n/a	Low	Low	Low	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.2	Capacity	High / Low	n/a	Low	Low	Low	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.3	Newness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.1.4	Predictability	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.5	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.1.6	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
				5.1.7	Social Orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		5.2	Situational Awareness Synthesis	5.2.1	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.2.2	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.2.3	Social orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.4	Assess implications	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		5.3	Decision Making	5.3.1	Risk Tolerance	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.2	Time variance	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.3	Decide	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.3.4	Autonomy	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		5.4	Error Management	5.4.1	Retrieval	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.4.2	Cognitive preparedness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				5.4..3.	Flexibility in action	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
6	Deployability	6.1	Mobility	6.1.1	Footprint	Small / Medium / Large	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small
				6.1.2	Endurance	Short / Medium / Long	Short	Short	Medium	Short	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL	Short
				6.1.3	Range	Short / Medium / Long	Medium	Medium	Long	Medium	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL	Medium
		6.2	Recoverability			High /	High	High	High	High	Low	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
						Low																
7	Effectiveness	7.1	Accessibility			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.2	Configurability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		7.3	Dependability			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.4	Integrity			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.5	Lethality	7.5.1	Deadliness	High / Low	High	Low	High	High	Low	High	High	High	High	High	High	High	High	FAIL	High	High
		7.6	Mobility	6.1.1	Footprint	Small / Medium / Large	Small	small	small	small	small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small	Small
				6.1.2	Endurance	Short / Medium / Long	Short	Short	Medium	Short	Long	Short	Short	Short	Short	Short	Short	Short	Short	Short	FAIL	Short
				6.1.3	Range	Short / Medium / Long	Medium	Medium	Long	Medium	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	FAIL	Medium
		7.7	Awareness	7.7.1	Bandwidth	High / Low	n/a	Low	Low	Low	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.2	Capacity	High / Low	n/a	Low	Low	Low	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.3	Newness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.4	Predictability	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.7.5	Physical orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.6	Cognitive orientation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.7.7	Social Orientation	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		7.8	Timeliness			High /	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
						Low																
		7.9	Usability			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		7.10	Processing Capability	7.10.1	Speed	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.2	Ambiguousness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.3	Complexity	High / Low	n/a	Low	Low	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				7.10.4	Contradictions	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.5	Uncertainty (Value)	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.6	Incompleteness (Quality)	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.7	Objectivity	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.8	Prioritisation	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				7.10.9	Sharing	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
8	Flexibility	8.1	Configurability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		8.2	Interchangeability			High / Low	n/a	Low	Low	Low	Low	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
9	Interoperability	9.1	Compatibility	9.1.1	Commonality	High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		9.2	Sharability	9.2.1	Ambiguity	High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
10	Orientability					High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
11	Survivability	11.1	Resilience	11.1.1	Stability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				11.1.2	Durability	Short / Medium	Medium	Medium	Medium	Medium	Long	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium	Medium

ID(1)	Attribute	ID(2)	Secondar2y Attribute	ID(3)	Factor	*Measures*	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre (RCC)	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
						/ Long																
		11.2	Robustness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		11.3	Security			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		11.4	Susceptibility			High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
		11.5	Vulnerability			High / Low	n/a	High	High	High	High	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
12	Sustainability	12.1	Degradability			High / Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low	Low
		12.2	Dependability	12.2.1	Trustworthiness	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
		12.3	Resilience	12.3.1	Stability	High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High
				12.3.2	Durability	High / Low	Long	Medium	Medium	Medium	Long	FAIL	FAIL	FAIL	FAIL	Long	Long	Long	Long	Long	Long	Long
		12.4	Robustness			High / Low	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High	High

J.4 Physical Environment Profile

Table 27 – Physical Environment Profile

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	ID(4)	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
1	Climate (Köppen climate classification)	1.1	GROUP A: Tropical/mega thermal climates	1.1.2	Tropical rain forest climate		n/a	Y	y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.1.3	Tropical monsoon climate		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.1.4	Tropical wet and dry or savannah climate		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		1.2	GROUP B: Dry (arid and semiarid) climates				Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		1.3	GROUP C: Temperate/mesothermal climates	1.3.1	Mediterranean climates		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.3.2	Humid subtropical climates		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.3.3	Maritime Temperate climates or Oceanic climates		n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.3.4	Maritime Subarctic climates or Sub polar Oceanic climates		n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		1.4	GROUP D: Continental/micro	1.4.1	Hot Summer Continental		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	ID(4)	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
			thermal climate		climates																	
				1.4.2	Warm Summer Continental or Hemiboreal climates		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.4.3	Continental Subarctic or Boreal (taiga) climates		n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.4.4	Continental Subarctic climates with extremely severe winters		n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		1.5	GROUP E: Polar climates	1.5.1	Tundra climate		n/a	N	N	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				1.5.2	Ice Cap climate		n/a	N	N	N	N	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
2	Environment	2.1	Land	2.1.1	Relief	2.1.1.1	n/a	N	N	N	N	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.1.1.2	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.1.1.3	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				2.1.2	Vegetation	2.1.2.1	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.1.2.2	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.1.2.3	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
						2.1.2.4	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.1.2.5	n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		2.2	Sea	2.2.1	Sea Conditions	2.2.1.1	n/a	4-5?	4-5?	4-5?		FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
						2.2.1.2	n/a	ALL	ALL	ALL		FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		2.3	Littoral				n/a	Y	Y	Y		FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
3	Predictability	3.1	Predictable				n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		3.2	Unpredictable				Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		3.3	Static				n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		3.4	Changing				Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y

ID(1)	Group	ID(2)	Subgroup	ID(3)	Element	ID(4)	Mission	MH-53J 'Pave Low' Helicopter	F-15C 'Eagle' Tactical Fighter Aircraft	Sandy' A-10A Close Air Support Aircraft	USAF Rescue Co-ordination Centre	Systems - 1,2	Systems - 1,2,3	Systems - 1,3	Systems - 2,3	Systems - 1,2,3,4	Systems - 1,2,4	Systems - 1,3,4	Systems - 2,3,4	Systems - 1,4	Systems - 2,4	Systems - 3,4
4	Strategic Purpose	4.1	Offensive				Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
		4.2	Defensive				n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		4.3	OOTW				n/a	Y	N	N	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
5	Political Environment	5.1	Local Hostility	5.1.1	Friendly		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.2	Benign		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.3	Low		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.1.4	Medium		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				5.1.5	High		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
		5.2	Political Stability	5.2.1	Despotism		Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y
				5.2.2	High		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.3	Medium		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.4	Low		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL
				5.2.5	Anarchy		n/a	Y	Y	Y	Y	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL	FAIL

Appendix K
Logical Modelling Procedure & Notations

K.1 Natural Language Notations

The following table outlines the natural language notations used in Logical Modelling (C. Dickerson 2008).

Table 28 – Logical Modelling Natural Language Notations

Word	*Capitalised*	*Italics*	*Bold*	*Underlined*	*Font Size*
Defined Term	✓		✓	✓	Normal
Key Words	✓	✓	✓		Normal
Other Words in definition			✓		Normal
Other Words not in definition					Small

K.1.1 Example

Consider a definition of *System*:

[redacted]

(International Council on Systems Engineering 2007)

In the context of engineering the term System will always mean a man-made system. Based on the INCOSE definition of System, the term would be comprised of 7 key words: combination, interacting, elements, organised, achieve, stated, and purpose.

K.2 Logical Assessment

Criteria for a logical model of a term include the intrinsic consistency and completeness of the model, and the correctness of the model relative to the natural language definition and under accepted interpretations. It is also desirable to have independence between the relations and independence between the key words.

K.3 Finishing the Procedure

The procedure is finished when all of the key words (both cited from the definition and necessary for completion of the model) have been identified and all of the relations between the words (both explicit and intended by the natural language definition) have been captured in the diagram.

K.4 Graphical Notations

The first step in modelling a term is to list out the key words that give it meaning. The key words will be undefined and their meaning will be determined by relations between the words, which will be represented graphically, using notations from the Unified Modelling Language (UML). The primary type of diagram needed from UML is the class diagram.

Using the diagram notation presented in Appendix K.1 the model is built starting from the keyword being defined (*system*) and identifying its direct relationship to other keywords. The identification process is iterated until all direct relationships between keywords (both explicit and intended by the natural language definition) have been identified. Figure 92 illustrates the result of this process for the seven keywords in *system*.

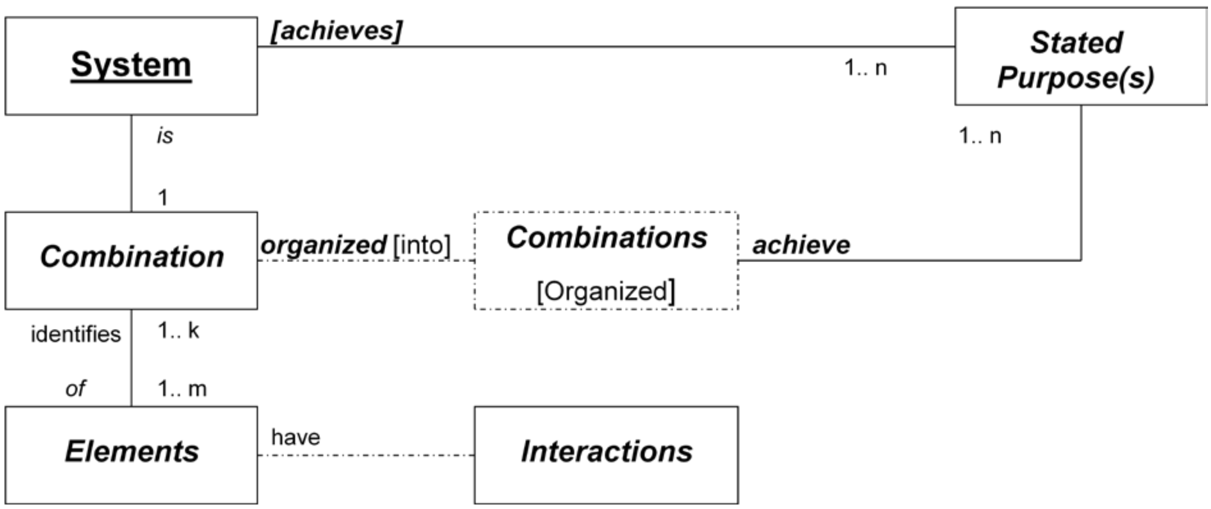


Figure 92 – Initial Logical Model for System

Appendix L
Model Transformations

This Appendix presents a more detailed explanation of the model transformations as discussed in section 5.4.2 and based on the work by Dickerson (2008).

The relational structure of a model is represented by a matrix, the notation of which is an underlined capital letter, such as $\underline{M}$. The notation for the underlying set of parameters is a capital italics letter, e.g. $M = \{y_1, y_2, \dots y_{m-1}, y_m\}$. A mathematical relation on the parameters M is denoted as a normal capital letter, for example $R = \{(y_2, y_{m-1}), (y_2, y_m), (y_{m-1}, y_2)\}$. The ordered pairs in R represent relationships between the parameters and correspond to the tick marks in the matrix $\underline{M}$, as illustrated in Figure 93, which is read row then column.

Consider the Matrix $\underline{M}$:

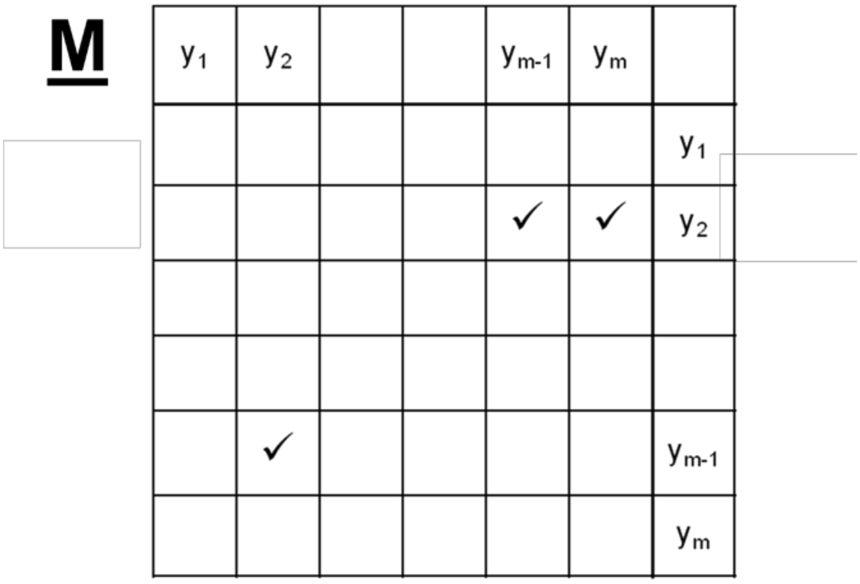


Figure 93 - Matrix $\underline{M}$

Relationships in $\underline{M}$: $\{(y_2, y_{m-1}), (y_2, y_m), (y_{m-1}, y_2)\}$

In Figure 94, Q transforms the parameters $y_1, \dots y_m$ into a set of parameters $x_1, \dots x_n$ by associating one or more of the x_k with one or more of the y_i. The notation y_iQx_k denotes that Q has associated y_i with x_k, hence the ordered pair (y_i, x_k) belongs to Q.

Q

x_1	x_2	x_3		x_{n-1}	x_n	
						y_1
	✓				✓	y_2
		✓				y_{m-1}
				✓		y_m

Figure 94 - Matrix Q

Relationships in Q: $\{(y_m, x_{n-1}), (y_{m-1}, x_3), (y_2, x_n), (y_2, x_2)\}$

The implied transformation of the models extends the notation yQx to the notation RQ to show how Q transforms a subset R of M into a subset RQ of N. Recall that a subset R of M, in general, represents a mathematical relation on the underlying set of parameters *M*. RQ is defined as follows: for each pair of parameters (y_i, y_j) that belong to the mathematical relation R, if y_iQx_k and y_jQx_l then the pair (x_k, x_l) belongs to the mathematical relation RQ in N. If S is a relation in N and RQ is a subset of S, then Q preserves the relationships of M and it is called a *relational transformation* from M to N. Although the transformation of relationships is easily calculated in this way, it is important to understand that this is not an ordinary calculation involving matrix sums or products. In summary from M and Q we have:

Relationships in M

- (y_2, y_{m-1})
- (y_2, y_m)
- (y_{m-1}, y_2)

Relationships in Q

- (y_m, x_{n-1})

- (y_{m-1}, x_3)
- (y_2, x_n)
- (y_2, x_2)

From Q the implications on the relationships in <u>M</u> are:

Implications of (y_2, y_{m-1})

- (y_2,x_n) & $(y_{m-1},x_3) \rightarrow (x_n,x_3)$
- (y_2,x_2) & $(y_{m-1},x_3) \rightarrow (x_2,x_3)$

Implications of (y_2,y_m)

- (y_2,x_n) & $(y_m,x_{n-1}) \rightarrow (x_n,x_{n-1})$
- (y_2,x_2) & $(y_m,x_{n-1}) \rightarrow (x_2,x_{n-1})$

Implications of (y_{m-1},y_2)

- (y_{m-1},x_3) & $(y_2,x_n) \rightarrow (x_3,x_n)$
- (y_{m-1},x_3) & $(y_2,x_2) \rightarrow (x_3,x_2)$

Hence, the relationships in <u>N</u> are: $\{(x_n,x_3), (x_2,x_3), (x_n,x_{n-1}), (x_2,x_{n-1}), (x_3,x_n), (x_3,x_2)\}$

Figure 95 illustrates the relational transformation described in this appendix.

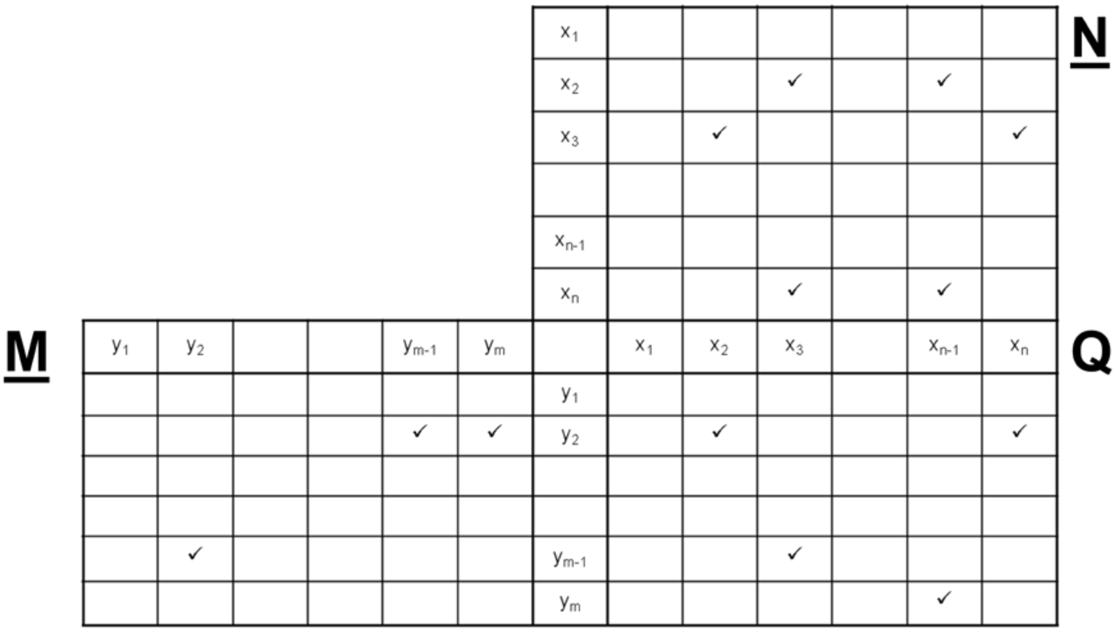


Figure 95 – Relational Transformation Q of <u>M</u> to <u>N</u>

Appendix M

Integrated Model Driven Approach

This Appendix presents a user manual for the use of the Integrated Model Driven Approach shown in section 5.5. For clarity the diagram shown previously in Figure 46 is repeated below in Figure 96 as it will be referred to throughout this Appendix.

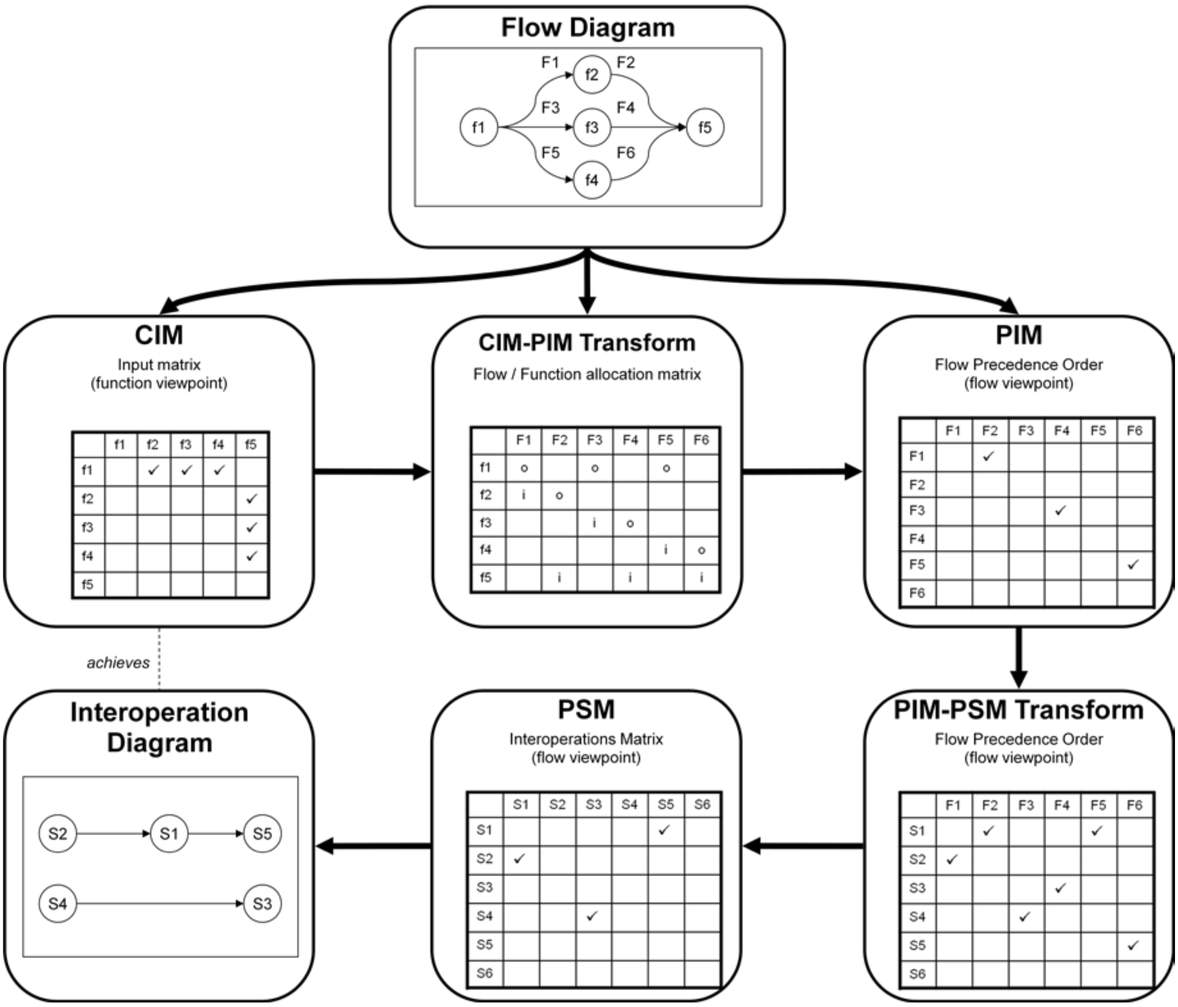


Figure 96 – Integrated Model Driven Approach

Each stage of the Integrated Model Driven Approach shown in Figure 96 is discussed in sequence in the following subsections. The purpose of this discussion is to serve as a guide; hence the specific terminology of CSAR will be abstracted up to more generic military missions. The first subsection provides an overview of the approach as a step by step guide for the user to follow. Subsequent subsections detail each of the steps.

M.1 Step by Step Guide

This subsection provides a step by step guide to the integrated Model Driven Approach shown in Figure 96.

1. Establish Flow Diagram
2. Determine CIM
3. Determine CIM-PIM Transform
4. Determine PIM
5. Complete PIM-PSM Transform
6. Transform PIM into PSM
7. Determine Interoperation Diagram

M.2 Flow Diagram

The input to the Integrated Model Driven Approach is the Flow Diagram shown at the top of Figure 96. The Flow Diagram captures a common model for the mission being conducted. This Flow Diagram represents the clear vision of what is required by a mission as expressed as processes and flows. As done in this thesis a precursor to producing the Flow Diagram may be the creation of a Logical Model for the mission in question. This Logical Model would be produced from available doctrines and tactics and its purpose is to capture a precise, coherent model of what the mission entails. Logical Modelling was discussed in the previous Appendix K. Note that this logical modelling approach will analyse the logical integrity of the doctrines and help expose any omission, inconsistencies and conflicts within them. The principle value of such an approach is the clear graphical exposure of the content of the doctrine which in turn can be used as a precise reference for the debate and resolution of any found issues. In practice this would require an experienced practitioner to analyse available doctrines to produce an initial model. This could then be refined through focus group discussion facilitated by the practitioner. This will be most effective when the subject matter experts are free to discuss and debate the integrity and completeness of the model and guide the facilitator to do the modelling. Hence, the subject matter experts are not encumbered by the technicalities of the techniques (which they would only briefly use) and can focus on the quality of the end product – the Flow Diagram. The people who help capture the mission in the way will be referred to as the mission architects.

M.3 CIM

The CIM can be determined from the Flow Diagram by considering the relationships between processes which is captured as a matrix. The resultant CIM helps identify the role of each process in terms of the number of interconnections. This helps reveal the relative importance of each of the processes in terms of mission importance by the dependencies as shown through the number of relationships that the process has. This would be used by the mission architects.

M.4 CIM-PIM Transform

The CIM-PIM Transform can be determined from the Flow Diagram by considering the relationship between processes and flows. This provides another view of the Flow Diagram and captures the process to flow relationships as a matrix. This helps visualise the numbers of flows associated with each process and helps show the importance of each. This would be used by the mission architects.

M.5 PIM

The CIM can be determined from the Flow Diagram by considering the precedent order of processes shown. This will require some interpretation as the precedents may become more complex as the scale and scope of the Flow Diagram increases. The PIM is the final artefact within the mission architects remit.

M.6 PIM-PSM Transform

The PIM-PSM Transform is completed by the decision maker in the field. This captures the relationship between available systems and the flows established in the PIM. This would be completed by the decision maker who would decide which particular systems they wished to use and the relationship of those systems to the flows established in the PIM. The decision maker only includes the systems they wish to use as a particular SoS, not all of the available systems. The decision maker may complete multiple PIM-PSM Transforms in order to compare these alternatives.

M.7 PSM

The PSM is the model of the specific SoS that will be employed for the mission and is generated from the PIM transformed through the PIM-PSM Transformation previously established. The PSM shows the relationships between the systems as a matrix. This helps the decision maker identify the interconnections within the SoS and helps them visualise the structure of the SoS. Where the decision maker has completed multiple PIM-PSM Transforms each will generate a PSM and these can be compared by the decision maker.

M.8 Interoperation Diagram

The final output of the Integrated Model Driven Approach is an Interoperation Diagram which is determined from the PSM. The Interoperation Diagram is the end point of this process as it shows the solution in a form that can be utilised by Architecture Frameworks such as the Ministry of Defence Architectural Framework (MODAF) and the Department of Defence Architecture Framework (DoDAF).

M.9 Further Thoughts

This Appendix has outlined how the Integrated Model Driven Approach may be applied in practice outside of this research domain of CSAR. The approach may be extended through automation. For example, the relationships between systems and flows may be predetermined by suitable subject matter experts and hence the decision maker may only need to select a set of systems to use – the PIM-PSM Transform being auto filled from their selection. The aim of this is to minimise the workload on the decision maker whilst maximising the output of the mission architects.

Appendix N

Model Transformation Software Tool

This Appendix presents the code written for the model transformation software tool implemented in Microsoft® Excel® 2007.

N.1 Description

This code was written in Microsoft Visual Basic and allows the transformation of a Matrix M into a Matrix N through transform Q. With the values for M and Q entered the values for N will be programmatically calculated.

N.2 Software Code

```
Sub FindRelationships()

' Macro written by Philip Johnson

' All variables declared

Dim Var1 As Variant
Dim Var2 As Variant
Dim Var3 As Variant
Dim CellRow As Variant
Dim CellColumn As Variant

Dim MArrayRow As Variant
Dim QArrayRow As Variant
Dim QArrayRow2 As Variant
Dim NArrayRow As Variant
Dim RowArrayRow As Variant
Dim ColumnArrayRow As Variant

Dim MArray(1 To 99, 1 To 2) As Variant
Dim QArray(1 To 99, 1 To 2) As Variant
Dim NArray(1 To 99, 1 To 2) As Variant
Dim RowArray(1 To 99, 1 To 2) As Variant
Dim ColumnArray(1 To 99, 1 To 2) As Variant

' Variables set to starting values

Var1 = 0
Var2 = 0
CellColumn = 1
CellRow = 1
MArrayRow = 1

' List M Relationships

CellRow = 1
CellColumn = 1
MArrayRow = 1

Do While CellRow <> 7
    Var1 = Worksheets("Sheet1").Cells(CellRow + 17, 10).Value
    Do While CellColumn <> 7
        Var2 = Worksheets("Sheet1").Cells((CellRow + 17), (CellColumn + 3)).Value
        If (IsEmpty(Var2)) Then
            CellColumn = CellColumn + 1
        Else
```

```
            MArray(MArrayRow, 1) = Var1
            MArray(MArrayRow, 2) = Worksheets("Sheet1").Cells(17, (CellColumn + 3)).Value
            MArrayRow = MArrayRow + 1
            CellColumn = CellColumn + 1
        End If
    Loop
    CellColumn = 1
    CellRow = CellRow + 1
Loop

' List Q Relationships

CellRow = 1
CellColumn = 1
QArrayRow = 1

Do While CellRow <> 7
    Var1 = Worksheets("Sheet1").Cells(CellRow + 17, 10).Value
    Do While CellColumn <> 7
        Var2 = Worksheets("Sheet1").Cells((CellRow + 17), (CellColumn + 10)).Value
        If (IsEmpty(Var2)) Then
            CellColumn = CellColumn + 1
        Else
            QArray(QArrayRow, 1) = Var1
            QArray(QArrayRow, 2) = Worksheets("Sheet1").Cells(17, (CellColumn + 10)).Value
            QArrayRow = QArrayRow + 1
            CellColumn = CellColumn + 1
        End If
    Loop
    CellColumn = 1
    CellRow = CellRow + 1
Loop

' Determine N Relationships

Var1 = 0
Var2 = 0
Var3 = 0

MArrayRow = 1
QArrayRow = 1
QArrayRow2 = 1
NArrayRow = 1

Do While MArray(MArrayRow, 1) <> Empty
    Var1 = MArray(MArrayRow, 1)
    Var2 = MArray(MArrayRow, 2)
    Do While QArray(QArrayRow, 1) <> Empty
        If Var1 = QArray(QArrayRow, 1) Then
            Var3 = QArray(QArrayRow, 2)
            Do While QArray(QArrayRow2, 1) <> Empty
                If Var2 = QArray(QArrayRow2, 1) Then
                    NArray(NArrayRow, 1) = Var3
                    NArray(NArrayRow, 2) = QArray(QArrayRow2, 2)
                    NArrayRow = NArrayRow + 1
                End If
                QArrayRow2 = QArrayRow2 + 1
            Loop
            QArrayRow2 = 1
        End If
        QArrayRow = QArrayRow + 1
    Loop
    QArrayRow = 1
    MArrayRow = MArrayRow + 1
Loop

' Change all cell shading colour to white

CellRow = 1
CellColumn = 1
Do While CellColumn <> 7
    Do While CellRow <> 7
        Worksheets("Sheet1").Cells(CellRow + 10, CellColumn + 10).Interior.ColorIndex = 2
        CellRow = CellRow + 1
    Loop
    CellRow = 1
    CellColumn = CellColumn + 1
Loop
```

```
' Shade Cells To Show Relationships
' Scan the row headings

CellRow = 1
RowArrayRow = 1

Do While CellRow <> 7
    Var1 = Worksheets("Sheet1").Cells(10 + CellRow, 17).Value
    If (IsEmpty(Var1)) Or Var1 = 0 Then
            CellRow = CellRow + 1
        Else
            RowArray(RowArrayRow, 1) = Var1
            RowArray(RowArrayRow, 2) = 10 + CellRow
            RowArrayRow = RowArrayRow + 1
            CellRow = CellRow + 1
    End If
Loop

' Scan the column headings

CellColumn = 1
ColumnArrayRow = 1

Do While CellColumn <> 7
    Var1 = Worksheets("Sheet1").Cells(17, 10 + CellColumn).Value
    If (IsEmpty(Var1)) Or Var1 = 0 Then
            CellColumn = CellColumn + 1
        Else
            ColumnArray(ColumnArrayRow, 1) = Var1
            ColumnArray(ColumnArrayRow, 2) = 10 + CellColumn
            ColumnArrayRow = ColumnArrayRow + 1
            CellColumn = CellColumn + 1
    End If
Loop

' Match data to row and column headings

CellRow = 1
CellColumn = 1
NArrayRow = 1

Do While NArray(NArrayRow, 1) <> Empty
    Var1 = NArray(NArrayRow, 1)
    Var2 = NArray(NArrayRow, 2)

    RowArrayRow = 1
    Do While RowArray(RowArrayRow, 1) <> Empty
    If Var1 = RowArray(RowArrayRow, 1) Then
        CellRow = RowArray(RowArrayRow, 2)
        Exit Do
        Else
        RowArrayRow = RowArrayRow + 1
        End If
    Loop

    ColumnArrayRow = 1
    Do While ColumnArray(ColumnArrayRow, 1) <> Empty
    If Var2 = ColumnArray(ColumnArrayRow, 1) Then
        CellColumn = ColumnArray(ColumnArrayRow, 2)
        Exit Do
        Else
        ColumnArrayRow = ColumnArrayRow + 1
        End If
    Loop


    If Worksheets("Sheet1").Cells(CellRow, CellColumn).Value = "ü" Then
        Worksheets("Sheet1").Cells(CellRow, CellColumn).Interior.ColorIndex = 12
        Else
        Worksheets("Sheet1").Cells(CellRow, CellColumn).Interior.ColorIndex = 3
    End If
    NArrayRow = NArrayRow + 1
Loop

End Sub
```

Appendix O

Executable Architectures – Air Warfare Model

[redacted]

O.1 Air Warfare System Concept Statement

[redacted]

[redacted]

O.2 Air Warfare System

[redacted]

Figure 97 – The Air Warfare System

[redacted]

[redacted]

Figure 98 – The Gas Station System

[redacted]

Table 29 – Interpretation of Gas Station Variables to Air Warfare Variables

[redacted]

[redacted]

O.3 Executable Architectures

[redacted]

[redacted]

O.4 Partitioning Systems

[redacted]

[redacted]

Figure 99 – Functional Versus Domain Partitioning

[redacted]

O.5 Domain

[redacted]

[redacted]

Figure 100 – Air Warfare Domain Model

[redacted]

[redacted]

Figure 101 – Air Warfare Domain Model with Viewpoints

[redacted]

O.6 The Implementation of xUML within Kennedy Carter's iUMLite

[redacted]

[redacted]

Figure 102 – xUML Model Layers

[redacted]

[redacted]

Figure 103 – Executable UML (xUML)

[redacted]

O.7 A Practical Example of Reusing Executable Architectures with iUMLite

[redacted]

[redacted]

Figure 104 – The Role of Sequence Diagrams

[redacted]

[redacted]

Figure 105 – Use Case: Threat Engagement and Weapon Delivery

[redacted]

[redacted]

Figure 106 – Sequence Diagram: Air Warfare System Engages Threat

[redacted]

[redacted]

Figure 107 – Class Diagram: Air Warfare Control

[redacted]

Figure 108 – Class Collaboration Diagram: Air Warfare Control

[redacted]

[redacted]

Figure 109 – Example State Machine ASL Code Translation

[redacted]

O.8 Simulating the Air Warfare Architecture

[redacted]

[redacted]

[redacted]

Figure 110 – iUMLite Simulator Screen Shot

[redacted]

[redacted]

O.8.1 Towards Future Executable Architectures

[redacted]

Appendix P
Translation Of iUMLite Gas Station Case Study To Air Warfare

This Appendix details the code translation of the iUMLite Gas Station case study to air warfare as outlined in the previous Appendix O

P.1 Introduction

We have taken the Gas Station case study from the book "Model Driven Architecture with Executable UML" and translated it in terms of a military application using the following platform specific model from the book "Using Architectures for Research Development and Acquisition."

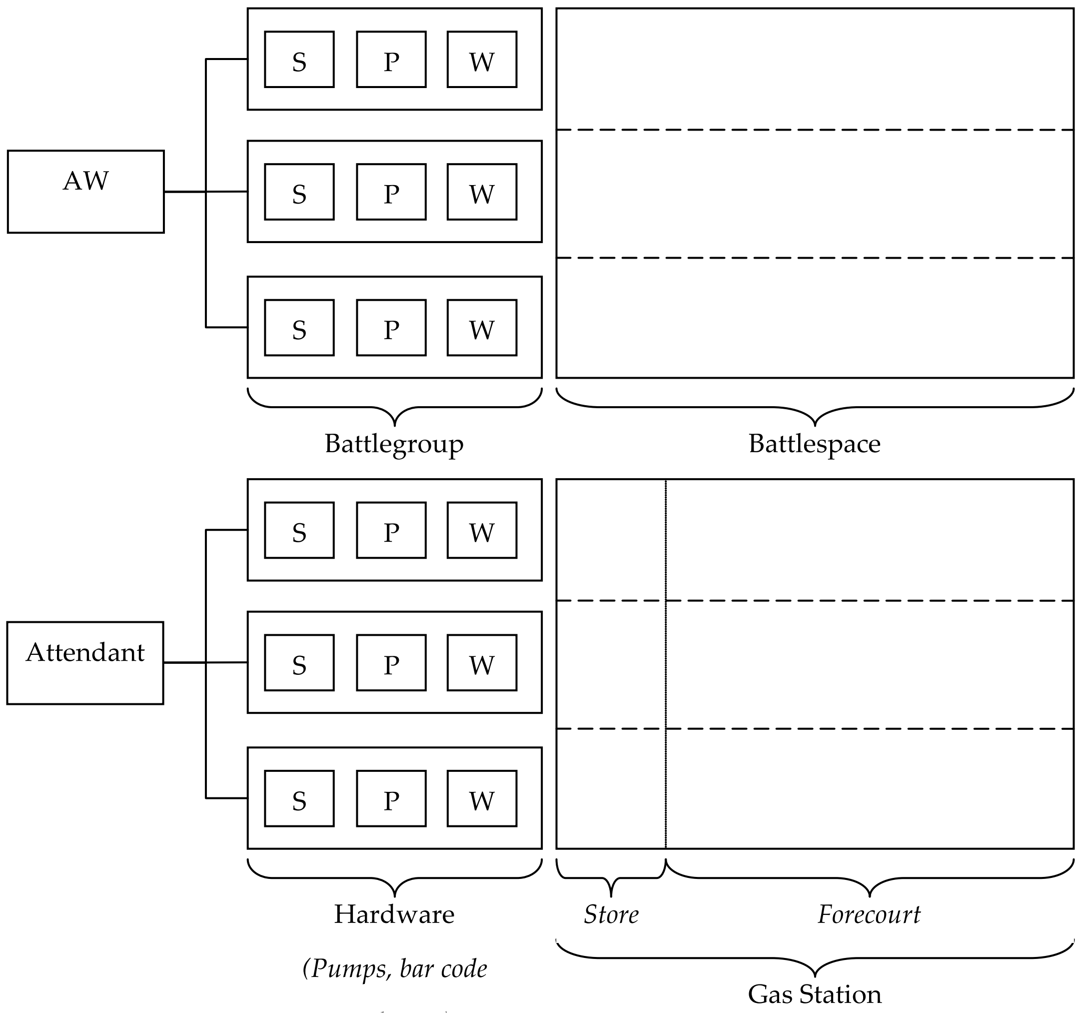


Figure 111 – Representations of Air Warfare and Gas Station Systems

P.2 Approach

1. Understand conceptual translation from the original Gas Station into an Air Warfare System
2. Map these translations between these two domains
3. Translate the Domain model within iUMLite (Use Cases & Sequence Diagrams) using the above map
4. Translate the Static Model within iUMLite (Classes, Associations and Generalisations) using the above map & translated Domain model.

P.3 Key Translation Map:

Note: This does not contain all translations; it is a list of the key translations made.

Customers ⇔ Threats

Attendant ⇔ AW: processing information; control

Forecourt ⇔ Battlespace: has threats [Responsive decisions, OODA]

Store ⇔ Near-in portion of Battlespace (point defence) [Reactive decisions, SCA]

Gas Station ⇔ "Area of influence" of Battlegroup within sensor & weapon range

Fuel ⇔ Weapons

Gas Pumps ⇔ Platforms (of Battlegroup)

Pump ⇔ Release Weapons (the act of firing weapons from the Weapon Launcher)

Delivery ⇔ Engagement (an event where a Platform prepares to engage a Threat)

Holster ⇔ Sensors (on Platform, e.g. RADAR)

Motor ⇔ Weapon System (the overall weapon system, include Tracking etc)

Clutch ⇔ Weapon Launcher (the actual launch mechanism for the weapon)

Transaction ⇔ D3 Assessment (analysis of effectiveness of an Engagement)

Delivery ⇔ Engagement

Gun ⇔ Trigger (fires Weapon Launcher)

Fuel Grade ⇔ Weapon Type

Fuel Grade (Four Star) ⇔ Weapon Type(Aster) – Aster is a surface to air missile

Fuel Grade (Unleaded) ⇔ Weapon Type(Rapier) – Rapier is a surface to air missile

P.4 Key Variable Translations List

Note: This does not contain all variable translations; it is a list of the key ones made.

connected_pumps = connected_platforms

Cost = Cost

Current_State = Current_State

Delivered_Volume = Engaged_Units

Delivery_Cost = Engagement_Cost

delivery_pump = engagement_platform

Delivery_Pump = Engagement_Platform

Delivery_Start_Time = Engagement_Start_Time

Delivery_Time = Engagement_Time

Fuel_Available = Weapons_Available

new_pending_transaction = new_pending_d3assessment

new_transaction = new_d3assessment

new_transaction.Transaction_Number = new_d3assessment.D3Assessment_Number

'Pending' = 'Pending'

PENDING_TRANSACTION = PENDING_D3ASSESSMENT

PUMP = PLATFORM

pump = platform

Pump_Number = Platform_Number

this.Tank_Capacity = this.Store_Level

this.Tank_Level = this.Store_Level

TRANSACTION = D3ASSESSMENT

Transaction_Number = D3Assessment_Number

Transaction_Type = D3Assessment_Type

Volume_Delivered=Engaged_Units

P.5 Code Translation

The following sections contain code translation from within iUMLite which have been captured as a resource for future coding activities. Original code form the Gas Station model is shown in italics and the translated code is shown in normal font.

P.6 D3 ASSESSMENT (TRN)

(Orig. TRANSACTION)
Transactions record the interaction between a customer and a pump that has resulted in fuel being pumped.

The transaction is identified by a unique arbitrary number. It is envisaged that transactions shall be archived at the end of each working day.

The transaction records a number of details including The Pump used, Cost, Delivery Start time and the time the transaction was last processed.

There are also a number of types of transaction :-
'Evaded Transaction' - The customer absconds

'Paid Transaction' - The customer has paid for the fuel
'Pending Transaction' - Waiting for the customer to pay.
Transactions record the interaction between a customer and a pump that has resulted in fuel being pumped.

D3Assessments record the interaction between a threat and a platform that has resulted in weapons being released.

The d3assessment is identified by a unique arbitrary number. It is envisaged that d3assessments shall be archived at the end of each day.

The d3assessment records a number of details including The Platform used, Cost, Engagement Start time and the time the d3assessment was last processed.

There are also a number of types of d3assessments :-
'Evaded D3Assessment' - The threat escapes
'Neutralized D3Assessment' - The threat was neutralized
'Pending D3Assessment' - Waiting for a d3assessment of the threat.

P.6.1 State Machine

1 Created

(Orig. Created)

3 Neutralized

(Orig. Paid)
Migrate pending subtype to paid subtype. Link all payments to the paid subtype.
 # Calculate the number of tokens and tell the operator to issue them to
 # the customer.

```
switch this.Transaction_Type
case 'Pending'
        pending_transaction = this->R4.PENDING_TRANSACTION
        unlink this R4 pending_transaction
        connected_pump = pending_transaction->R9
        unlink pending_transaction R9 connected_pump
        delete pending_transaction
case 'Evaded'
        evaded_transaction = this->R4.EVADED_TRANSACTION
        unlink this R4 evaded_transaction
        delete evaded_transaction
endswitch

paid_transaction = create PAID_TRANSACTION with \
        Transaction_Number = this.Transaction_Number
this.Transaction_Type = 'Paid'

link this R4 paid_transaction
```

Migrate pending subtype to neutralized subtype. Link all d3assessments to the neutralized subtype.
Calculate the number of tokens and tell the operator to issue them to
the customer.

```
switch this.D3Assessment_Type
case 'Pending'
	pending_d3assessment = this->R4.PENDING_D3ASSESSMENT
	unlink this R4 pending_d3assessment
	connected_platform = pending_d3assessment->R9
	unlink pending_d3assessment R9 connected_platform
	delete pending_d3assessment
case 'Evaded'
	evaded_d3assessment = this->R4.EVADED_D3ASSESSMENT
	unlink this R4 evaded_d3assessment
	delete evaded_d3assessment
endswitch

neutralized_d3assessment = create NEUTRALIZED_D3ASSESSMENT with \
	D3Assessment_Number = this.D3Assessment_Number
this.D3Assessment_Type = 'Neutralized'

link this R4 neutralized_d3assessment
```

4 Evaded

(Orig. Evaded)

 # Migrate pending subtype to evaded subtype.
 # Record customer details.

pending_transaction = this->R4.PENDING_TRANSACTION
unlink this R4 pending_transaction
pump = pending_transaction->R9
unlink pending_transaction R9 pump
delete pending_transaction

*evaded_transaction = create EVADED_TRANSACTION with *
*Transaction_Number = this.Transaction_Number & *
Observations = Customer_Details

this.Transaction_Type = 'Evaded'

link this R4 evaded_transaction

```
# Migrate pending subtype to evaded subtype.
# Record customer details.

pending_d3assessment = this->R4.PENDING_D3ASSESSMENT
unlink this R4 pending_d3assessment
platform = pending_d3assessment->R9
unlink pending_d3assessment R9 platform
delete pending_d3assessment

evaded_d3assessment = create EVADED_D3ASSESSMENT with \
	D3Assessment_Number = this.D3Assessment_Number & \
	Observations = Target_Details
```

this.D3Assessment_Type = 'Evaded'

link this R4 evaded_d3assessment
TRN1:Create D3Assessment
(Orig. Create Transaction)

 # Create pending transaction waiting payment from customer

```
new_transaction = create unique TRANSACTION with \
        Delivery_Start_Time = Delivery_Time & \
        Pump_Number = Delivery_Pump & \
        Cost = Delivery_Cost & \
        Transaction_Type = 'Pending' & \
    Current_State = 'Created'

new_pending_transaction = create PENDING_TRANSACTION with \
        Transaction_Number = new_transaction.Transaction_Number

link new_transaction R4 new_pending_transaction

delivery_pump = find-one PUMP where Pump_Number = Delivery_Pump

link new_transaction R10 delivery_pump
link new_pending_transaction R9 delivery_pump

# Create pending d3assessment whilst waiting d3 assessment of threat

new_d3assessment = create unique D3ASSESSMENT with \
        Engagement_Start_Time = Engagement_Time & \
        Platform_Number = Engagement_Platform & \
        Cost = Engagement_Cost & \
        D3Assessment_Type = 'Pending' & \
    Current_State = 'Created'

new_pending_d3assessment = create PENDING_D3ASSESSMENT with \
        D3Assessment_Number = new_d3assessment.D3Assessment_Number

link new_d3assessment R4 new_pending_d3assessment

engagement_platform = find-one PLATFORM where Platform_Number = Engagement_Platform

link new_d3assessment R10 engagement_platform
link new_pending_d3assessment R9 engagement_platform
```

P.6.2 TRN2:D3 Assessment

(Orig. Payment Received)

P.6.3 TRN3:Threat Escapes

(Orig. Customer Absconds)

P.6.4 TRN4:Threat Neutralized

(Orig. Transaction Paid)

P.7 ENGAGEMENT (EGT)

(Orig. Delivery)

An Engagement records the Threat interaction with a Platform.

The Engagement is identified by the Platform Number and the time the weapon engagement started. It also records the units of weapons released and the cost.

On completion of the Engagement if the Threat has been engaged with weapons then a D3 Assessment is created with the appropriate details. If no weapons were engaged then the engagement is deleted and no D3 Assessment created.

P.7.1 State Machine

3 Creating Engagement

(Orig. Creating Delivery)

 # Create a delivery instance and enable the pump.

time_now = current-time

```
new_delivery = create DELIVERY with \
        Pump_Number = Pump_Id & \
        Time = time_now & \
        Volume_Delivered = 0.0 & \
        Cost = 0.0

delivering_pump = find-one PUMP where Pump_Number = Pump_Id

link new_delivery R3 delivering_pump

generate PMP7:Pump_Enabled() to delivering_pump

# Create an engagement instance and enable the platform.

time_now = current-time

new_engagement = create ENGAGEMENT with \
        Platform_Number = Platform_Id & \
        Time = time_now & \
        Engaged_Units = 0.0 & \
        Cost = 0.0

engaging_platform = find-one PLATFORM where Platform_Number = Platform_Id

link new_engagement R3 engaging_platform

generate PTM7:Platform_Enabled() to engaging_platform
```

5 Calculating price of unconstrained engagement

(Orig. Calculating price of unconstrained delivery)

 # A unit of petrol has been delivered.
 # Update the volume delivered and current cost.

new_volume = this.Volume_Delivered + 0.01
this.Volume_Delivered = new_volume

grade = this->R3->R1->R2
new_cost = this.Cost + grade.Unit_Price
this.Cost = new_cost

A unit of weapons has been engaged.
Update the units delivered and current cost.

new_engaged_units = this.Engaged_Units + 1.0
this.Engaged_Units = new_engaged_units

weapon_type = this->R3->R1->R2
new_cost = this.Cost + weapon_type.Unit_Cost
this.Cost = new_cost

6 Engagement Complete

(Orig. Delivery Complete)

 # Delivery completed so create a transaction.
 # Inform the tank amount of fuel dispensed.

[] = TRN1:Create_Transaction[this.Time, this.Pump_Number, this.Cost]

supplying_tank = this -> R3 -> R1
generate TNK4:Fuel_Used(this.Volume_Delivered) to supplying_tank

pump = this->R3
unlink this R3 pump
delete this

Engagement completed so create a d3assessment.
Inform the weapon store amount of weapons engaged.

[] = TRN1:Create_D3Assessment[this.Time, this.Platform_Number, this.Cost]

supplying_weapon_store = this -> R3 -> R1
generate WPS4:Weapons_Used(this.Engaged_Units) to supplying_weapon_store

platform = this->R3
unlink this R3 platform
delete this

8 Engagement Cancelled

(Orig. Delivery Cancelled)

 # The customer has not pumped any fuel therefore there is
 # no pending transaction to be paid.

[] = AT3:Delivery_Cancelled[]

pump = this->R3
unlink this R3 pump
delete this

The no weapons engaged therefore there is
no pending d3assessment to be done.

[] = AW3:Engagement_Cancelled[]

platform = this->R3
unlink this R3 platform
delete this

P.7.2 EGT3:Create Engagement

(Orig. DEL3:Create Delivery)

P.7.3 EGT4:Weapon Unit Engaged

(Orig. DEL4:Fuel Unit Delivered)

P.7.4 EGT5:Engagement Complete

(Orig. DEL5:Delivery Complete)

P.7.5 EGT6:Delete Engagement

(Orig. DEL6:Delete Delivery)

P.8 PENDING D3ASSESSMENT (PND)

(Orig. Pending Transaction)

A D3 Assessment that is waiting for sensor data.

The Platform where the Engagement has been performed is recorded.

P.9 PLATFORM (PTM)

(Orig. PUMP)

Represents the Platform within the Battlespace.

Each platform is uniquely identified by a number.

Each platform supplies one type of weapon and is permanently connected to a weapon store supplying that Weapon Type.

P.9.1 State Machine

1 Waiting For Threat

(Orig. Waiting For Customer)

 # The pump is idle. Wait in this state until
 # a customer removes the gun from its holster.

The platform is idle. Wait in this state until
a threat is detected by sensors.

2 Waiting Platform Enable

(Orig. Waiting Platform Enable)

 # Determine whether the connected tank contains
 # more than 4% of its capacity.

```
supplying_tank = this -> R1

if supplying_tank.Tank_Empty_Flag = TRUE then
  generate PMP4:Fuel_Level_Low() to this
else
  [] = AT1:Request_Pump_Enable[]
endif
```

Determine whether the connected weapons store contains
more than 4% of its capacity.

```
supplying_weapon_store = this -> R1

if supplying_weapon_store.Weapon_Store_Empty_Flag = TRUE then
  generate PTM4:Weapon_Level_Low() to this
else
  [] = AW1:Request_Platform_Authorization[]
endif
```

5 Weapons Unavailable

(Orig. Fuel Unavailable)

 # Inform customer that the pump is unavailable.

 # Wait for fuel to become available for this pump.

[] = CU1:Pump_Unavailable[]

Inform weapons officer that the platform is unavailable.

Wait for weapons to become available for this platform.

```
[] = CU1:Platform_Unavailable[]
```

6 Weapon Engagement Complete
(Orig. Fuel Delivery Complete)

 # Delivery Now Complete, Stop motor and return pump
 # to waiting state

current_delivery = this->R3
generate DEL5:Delivery_Complete() to current_delivery

[] = MO2:Stop_Motor[]
generate PMP12:Customer_Finished() to this

Engagement Now Complete, Stop weapon system and return platform
to waiting state

current_engagement = this->R3
generate EGT5:Engagement_Complete() to current_engagement

[] = MO2:Stop_Weapon_System[]
generate PTM12:Engagement_Finished() to this

7 Ready To Release Weapons

(Orig. Ready To Pump)

 # Start pump motor and wait for the gun trigger to be depressed.
 # Or the gun can be replaced.

[] = MO1:Start_Motor[]

Start weapon system and wait for the trigger to be depressed.
Or the weapon system can be stopped.

[] = MO1:Start_Weapon_System[]

8 Weapon Release Paused

(Orig. Pumping Paused)

 # Disengage clutch which stops pumping.
 # Wait for gun to be replaced into the
 # holster or for the trigger to be depressed.

[] = CL2:Disengage_Clutch[]
Disengage weapon launcher which stops releasing weapons.
Wait for threat not to be detected
or for the trigger to be depressed.

[] = CL2:Disengage_Weapon_Launcher[]

11 Release Weapons

(Orig. Pumping)

 # Engage clutch which starts pumping.
 # Continue until the gun trigger is released

[] = CL1:Engage_Clutch[]

Engage weapon launcher which starts releasing weapons.
Continue until the trigger is released

[] = CL1:Engage_Weapon_Launcher[]

P.9.2 PTM1:Threat Detected

(Orig. PMP1:Pump Removed)

P.9.3 PTM2:Threat Not Detected

(Orig. PMP2:Gun Replaced)

P.9.4 PTM3:Weapon Level OK

(Orig. PMP3:Fuel Level OK)

P.9.5 PTM4:Weapon Level Low

(Orig. PMP4:Fuel Level Low)

P.9.6 PTM5:Weapons Available

(Orig. PMP5:Fuel Available)

P.9.7 PTM7:Platform Enabled

(Orig. PMP7:Pump Enabled)

P.9.8 PTM8:Trigger Depressed

(Orig. PMP8:Trigger Depressed)

P.9.9 PTM9:Trigger Released

(Orig. PMP9:Trigger Released)

P.9.10 PTM12:Engagement Finished

(Orig. PMP12:Customer Finished)

P.10TARGET ESCAPED (EVD)

(Orig. EVADED TRANSACTION)

A D3 Assessment where the target has escaped without D3.

P.11TARGET NEUTRALIZED (PDT)

(Orig. PAID TRANSACTION)

A D3 Assessment where the Target has been D3'd.

P.12WEAPON STORE (WPS)

(Orig. TANK)

Represents the storage area of weapons within the Battlespace. A Weapon Store contains one type of Fuel and cannot be re-filled with another fuel type.

Each weapon store a Capacity in units, current level in units, an empty threshold which is the percentage that the weapon store is allowed to fall to before it is regarded as being Empty.

Each weapon store is identified by a Unique number.

P.12.1 State Machine

1 Check Levels After Weapon Delivery
(Orig. Checking Levels After Tanker Delivery)

 # Add volume to tank level. If the level is greater
 # than 4% of the tanks capacity reset any waiting pumps.
 # If the tank is still below its threshold return to the
 # Nearly Empty state.

[] = TNK3:Increase_Level[Added_Volume] on this

[below_threshold] = TNK1:Check_Level[] on this

```
if below_threshold = TRUE then
        this.Tank_Empty_Flag = TRUE
        generate TNK2:Level_Below_Threshold() to this
else
        this.Tank_Empty_Flag = FALSE
        [] = TNK4:Inform_Connected_Pumps_Fuel_Available[] on this
        generate TNK3:Level_Above_Threshold() to this
endif
```

Add units to weapon store level. If the level is greater
than 4% of the weapon store capacity reset any waiting platforms.
If the weapon store is still below its threshold return to the
Nearly Empty state.

[] = WPS3:Increase_Level[Added_Units] on this

[below_threshold] = WPS1:Check_Level[] on this

```
if below_threshold = TRUE then
        this.Weapon_Store_Empty_Flag = TRUE
        generate WPS2:Level_Below_Threshold() to this
```

else
            this.Weapon_Store_Empty_Flag = FALSE
            [] = WPS4:Inform_Connected_Platforms_Weapons_Available[] on this
            generate WPS3:Level_Above_Threshold() to this
endif

2 Waiting For Weapon Delivery

(Orig. Waiting For Tanker Delivery)

 # Fuel tank level below 4%. Wait for fuel to be delivered.

Weapon Store level below 4%. Wait for weapons to be delivered.

3 Reset Waiting Platforms

(Orig. Reset Waiting Pumps)
 # Tank In Use with greater than 4% of volume

Weapon Store In Use with greater than 4% of volume

4 Checking Levels After Platform Usage

(Orig. Checking Levels After Pump Usage)

 # Reduce recorded tank level by volume delivered.
 # If the level is less than 4% of the tanks capacity
 # then inhibit connected pumps from making further
 # deliveries.

[] = TNK2:Reduce_Level[Delivered_Volume] on this

[below_threshold] = TNK1:Check_Level[] on this

if below_threshold then
            this.Tank_Empty_Flag = TRUE
            generate TNK2:Level_Below_Threshold() to this
else
            generate TNK3:Level_Above_Threshold() to this
endif

Reduce recorded weapon store level by units engaged.
If the level is less than 4% of the weapon stores capacity
then inhibit connected platforms from making further
engagements.

[] = WPS2:Reduce_Level[Engaged_Units] on this

[below_threshold] = WPS1:Check_Level[] on this

if below_threshold then
            this.Weapon_Store_Empty_Flag = TRUE
            generate WPS2:Level_Below_Threshold() to this
else

generate WPS3:Level_Above_Threshold() to this
endif

5 Updating Weapon Levels

(Orig. Updating Fuel Levels)

 # Fuel has been used so reduce the volume in the tank.

[] = TNK2:Reduce_Level[Delivered_Volume] on this

generate TNK3:Level_Below_Threshold() to this

Weapons have been engaged/used so reduce the number (units) in the tank.

[] = WPS2:Reduce_Level[Engaged_Units] on this

generate WPS3:Level_Below_Threshold() to this

P.12.2 WPS1: Check Level

(Orig. TNK1:Check Level)

*percent_level = (this.Tank_Level / this.Tank_Capacity) * 100.0*

if percent_level < this.Empty_Threshold then
Below_Threshold = TRUE
else
Below_Threshold = FALSE
endif

percent_level = (this.Weapon_Store_Level / this.Weapon_Store_Capacity) * 100.0

if percent_level < this.Empty_Threshold then
Below_Threshold = TRUE
else
Below_Threshold = FALSE
endif

P.12.3 WPS2:Reduce Level

(Orig. TNK2:Reduce Level)

new_level = this.Tank_Level - Delivered_Volume
this.Tank_Level = new_level

new_level = this.Weapon_Store_Level – Engaged_Units
this.Weapon_Store_Level = new_level

P.12.4 WPS3:Increase Level

(Orig. TNK3:Increase Level)

new_level = this.Tank_Level + Delivered_Volume *this.Tank_Level = new_level*

new_level = this.Weapon_Store_Level + Engaged_Units
this.Weapon_Store_Level = new_level

P.12.5 WPS4:Inform Connected Platforms Weapons Available

(Orig. TNK4:Inform Connected Pumps Fuel Available)

 # The fuel level in this tank has return to
 # an operational level. Inform all connected
 # pumps that the fuel level is ok

{connected_pumps} = this->R1

```
for pump in {connected_pumps} do
        generate PMP5:Fuel_Available() to pump
endfor
```

The weapons level in this store has returned to
an operational level. Inform all connected
platforms that the weapon level is ok

{connected_platforms} = this->R1

```
for platform in {connected_platforms} do
        generate PMP5:Weapons_Available() to pump
endfor
```

P.13WEAPON TYPE (WPE)

(Orig. FUEL GRADE)

Each Platform can release a number of different types of weapon. Each weapon type has a name which uniquely identifies that weapon type and a cost for each use of that weapon type.

P.14AW (AW)

(Orig. ATTENDANT)

Represents the Air Warfare Officer interface.

P.14.1 AW1:Request Platform Authorization

(Orig. AT1:Request Pump Enable)

P.14.2 AW2:D3 Assessment Pending

(Orig. AT2:Transaction Pending)

P.14.3 AW3: Engagement Cancelled

(Orig. AT3:Delivery Cancelled)

P.15WEAPON SYSTEM (MO)

(Orig. MOTOR)

Represents the interface with the Platform weapon system for targeting, tracking etc. Note that the actual release/launch of weapons is handled by the Weapon Launcher.

P.15.1 MO1:Start Weapon System

(Orig. MO1:Start Motor)

P.15.2 MO2:Stop Weapon System

(Orig. MO2:Stop Motor)

P.16WEAPONS OFFICER (CU)

(Orig. CUSTOMER)
Represents interface with weapons officer.

P.16.1 CU1: Platform Unavailable

(Orig. CU1:Pump Unavailable)

P.17METER (ME)

(Orig. METER)

Represents meter associated with a pump.
Represents meter associated with a Platform – tracks number of weapons launched in an Engagement.

P.18WEAPON LAUNCHER

(orig. CLUTCH)

Represents interface with pump's clutch.
Represents interface with Platform Weapon Launcher (the devices which launch the weapons).

P.18.1 CL1:Engage Weapon Launcher

(Orig. CL1:Engage Clutch)

P.18.2 CL2:Disengage Weapon Launcher

(Orig. CL2:Disengage Clutch)

P.19WEAPON LOGISTICS OFFICER (WLO)

(Orig. TANKER OPERATOR)

Represents interface with weapons logistics officer

P.20TRIGGER (TR)

(Orig. GUN)

Represents interface with Platforms weapon launcher.

P.21SENSORS (HO)

(Orig. HOLSTER)

Represents interface with pump's holster.

P.22Model Execution Environment

P.22.1 Initialisation Segment Method

```
# Populate Gas Station Control domain with the initial
# configuration. This station supplies two grades of fuel
# "Four Star" & "Unleaded". This station has two fuel
# storage tanks one containing "Four Star" the other
# "Unleaded". This station has three pumps
# pump 1 is supplied with "Four Star",  pumps 2 & 3 are
# supplied with "Unleaded"

# First instantiate 2 tanks which supply different fuel grades.

four_star_fuel_grade = create FUEL_GRADE with Grade_Name = "Four Star" \
                              & Unit_Price = 62.9

unleaded_fuel_grade = create FUEL_GRADE with Grade_Name = "Unleaded" \
                             & Unit_Price = 59.5

tank_1001 = create TANK with Tank_Number = 1001 \
                  & Tank_Empty_Flag = TRUE \
                  & Tank_Level = 0.0 \
                  & Tank_Capacity = 100000.0 \
                  & Empty_Threshold = 4.0 \
                  & Current_State = 'Waiting_For_Tanker_Delivery'

tank_1002 = create TANK with Tank_Number = 1002 \
                  & Tank_Empty_Flag = FALSE \
                  & Tank_Level = 10000.0 \
                  & Tank_Capacity = 200000.0 \
                  & Empty_Threshold = 4.0 \
                  & Current_State = 'Reset_waiting_pumps'

link four_star_fuel_grade R2 tank_1001
link unleaded_fuel_grade R2 tank_1002

# Create Pump 1 that supplies Four Star

pump_1 = create PUMP with Pump_Number = 1 \
                & Current_State = 'Waiting_For_Threat'

link pump_1 R1 tank_1001
```

```
# Create Pumps 2 & 3 that supply Unleaded

pump_2 = create PUMP with Pump_Number = 2 \
                & Current_State = 'Waiting_For_Customer'

pump_3 = create PUMP with Pump_Number = 3 \
                & Current_State = 'Waiting_For_Customer'

link pump_2 R1 tank_1002
link pump_3 R1 tank_1002

# Populate Air Warfare Control domain with the initial
# configuration. This station supplies two types of weapon
# "Aster" (a type of Surface to Air Missile) & "Rapier" (another type of Surface to Air # Missile). This
battlespace has two weapon stores
# one containing "Aster" the other
# "Rapier". This battespace has three platforms
# platform 1 is supplied with "Aster",  platforms 2 & 3 are
# supplied with "Rapier"

# First instantiate 2 weapon stores which supply different weapon types.

aster_weapon_type = create WEAPON_TYPE with Weapon_Type = "Aster" \
                             & Unit_Cost = 100000

rapier_weapon_type = create WEAPON_TYPE with Weapon_Type = "Rapier" \
                             & Unit_Cost = 50000

weapon_store_1001 = create WEAPON_STORE with Weapon_Store_Number = 1001 \
                  & Weapon_Store_Empty_Flag = TRUE \
                  & Weapon_Store_Level = 0.0 \
                  & Weapon_Store_Capacity = 100000.0 \
                  & Empty_Threshold = 4.0 \
                  & Current_State = 'Waiting_For_Weapon_Delivery'

weapon_store_1002 = create WEAPON_STORE with Weapon_Store_Number = 1002 \
                  & Weapon_Store_Empty_Flag = FALSE \
                  & Weapon_Store_Level = 10000.0 \
                  & Weapon_Store_Capacity = 200000.0 \
                  & Empty_Threshold = 4.0 \
                  & Current_State = 'Reset_Waiting_Platforms'

link aster_weapon_type R2 weapon_store_1001
link rapier_weapon_type R2 weapon_store_1002

# Create Platform 1 that can launch Aster weapons

platform_1 = create PLATFORM with Platform_Number = 1 \
                & Current_State = 'Waiting_For_Threat'

link platform_1 R1 weapon_store_1001
```

```
# Create Platforms 2 & 3 that supply Rapier weapons

platform_2 = create PLATFORM with Platform_Number = 2 \
            & Current_State = 'Waiting_For_Threat'

platform_3 = create PLATFORM with Platform_Number = 3 \
            & Current_State = 'Waiting_For_Threat'

link platform_2 R1 weapon_store_1002
link platform_3 R1 weapon_store_1002
```

P.23Test Methods

P.23.1 1 THREAT detected by Platform 2

(Orig. CUSTOMER Removes Gun From Pump 2)

```
# Customer removes gun from pump 2's holster

pump = find-one PUMP where Pump_Number = 2
if pump != UNDEFINED then
  generate PMP1:Gun_Removed() to pump
else

$INLINE
  printf("Gas Station Error: There is no instance of Pump number 2.\n");
$ENDINLINE

Endif


# Threat detected by Platform 2's sensors


platform = find-one PLATFORM where Platform_Number = 2
if platform != UNDEFINED then
  generate PTM1:Threat_Detected() to platform
else

$INLINE
  printf("Air Warfare Error: There is no instance of Platform number 2.\n");
$ENDINLINE

endif
```

P.23.2 2 AW Enables Platform 2

(Orig. ATTENDENT Enables Pump 2)

```
# Attendent enables Pump number 2

generate DEL3:Create_Delivery(2)
```

AW enables Platform number 2

generate EGT3:Create_Engagement(2)

P.23.3 3 WEAPONS OFFICER Presses Trigger At Platform 2

(orig. CUSTOMER Presses Trigger At Pump 2)

 # Customer presses Gun trigger on pump 2

pump = find-one PUMP where Pump_Number = 2

```
if pump != UNDEFINED then
  generate PMP8:Trigger_Depressed() to pump
else

$INLINE
  printf("Gas Station Error: There is no Instance of Pump number 2.\n");
$ENDINLINE

endif
```

Weapons Officer presses weapon launcher trigger on platform 2

platform = find-one PLATFORM where Platform_Number = 2

```
if platform != UNDEFINED then
  generate PTM8:Trigger_Depressed() to platform
else

$INLINE
  printf("Air Warfare Error: There is no Instance of Platform number 2.\n");
$ENDINLINE

endif
```

P.23.4 4 METER Engages Weapon Unit For Platform 2

(Orig. METER Delivers Fuel Unit For Pump 2)

 # The delivery needs to be informed about each metered fuel delivery

delivery = find-one DELIVERY where Pump_Number = 2

```
if delivery != UNDEFINED then
  generate DEL4:Fuel_Unit_Delivered() to delivery
else

$INLINE
  printf("Gas Station Error: No active delivery for Pump number 2.\n");
$ENDINLINE

endif
```

The engagement needs to be informed about each metered weapon engagement

engagement = find-one ENGAGEMENT where Platform_Number = 2

if engagement != UNDEFINED then
  generate EGT4:Weapon_Unit_Engaged() to engagement
else

$INLINE
  printf("Air Warfare Error: No active engagement for Platform number 2.\n");
$ENDINLINE

endif

P.23.5 5 WEAPONS OFFICER Releases Trigger At Platform 2

(Orig. CUSTOMER Releases Trigger At Pump 2)

 # Customer releases Gun trigger on pump 2

pump = find-one PUMP where Pump_Number = 2

if pump != UNDEFINED then
generate PMP9:Trigger_Released() to pump
else

$INLINE
printf("Gas Station Error: There is no instance of Pump number 2.\n");
$ENDINLINE
endif

Weapons Officer releases weapons launcher trigger on platform 2

platform = find-one PLATFORM where Platform_Number = 2

if platform != UNDEFINED then
  generate PTM9:Trigger_Released() to platform
else

$INLINE
  printf("Air Warfare Error: There is no instance of Platform number 2.\n");
$ENDINLINE
endif

P.23.6 6 WEAPONS OFFICER detects no Threat At Platform 2

(Orig. CUSTOMER Replaces Gun At Pump 2)

 # Customer replaces gun in pump 2's holster

pump = find-one PUMP where Pump_Number = 2

if pump != UNDEFINED then
  generate PMP2:Gun_Replaced() to pump
else

$INLINE
  printf("Gas Station Error: There is no instance of Pump number 2.\n");
$ENDINLINE

endif

Weapons Officer detects no Threat at Platform 2

platform = find-one PLATFORM where Platform_Number = 2

if platform != UNDEFINED then
  generate PTM2:Threat_Not_Detected() to platform
else

$INLINE
  printf("Air Warfare Error: There is no instance of Platform number 2.\n");
$ENDINLINE

endif

P.23.7 7 THREAT Neutralized D3 Assessment for weapons engaged from Platform 2

(Orig. CUSTOMER Pays For Fuel For Pump 2)

 # Customer pays for fuel for Pump 2

pump = find-one PUMP where Pump_Number = 2

if pump != UNDEFINED then
  current_pending_transaction = pump->R9

  if current_pending_transaction != UNDEFINED then
    current_transaction = current_pending_transaction->R4
    generate TRN2:Payment_Received() to current_transaction
  endif
else

$INLINE
  printf("Gas Station Error: There is no instance of Pump number 2.\n");
$ENDINLINE

endif

Threat Neutralized D3 Assessment for weapons engaged from Platform 2

platform = find-one PLATFORM where Platform_Number = 2

```
if platform != UNDEFINED then
  current_pending_d3assessment = platform->R9

  if current_pending_d3assessment != UNDEFINED then
        current_d3assessment = current_pending_d3assessment->R4
        generate TRN2:D3_ASSESSMENT() to current_d3assessment
  endif
else

$INLINE
  printf("Air Warfare Error: There is no instance of Platform number 2.\n");
$ENDINLINE

endif
```

P.23.8 8 THREAT Escapes From Platform 2

(Orig. CUSTOMER Absconds From Pump 2)

```
# Customer absconds from Pump 2 without paying

pump = find-one PUMP where Pump_Number = 2

if pump != UNDEFINED then

  current_pending_transaction = pump -> R9

  if current_pending_transaction != UNDEFINED then
    current_transaction = current_pending_transaction -> R4
    generate TRN3:Customer_Absconds("Dodgy looking geezer in vehicle with registration number M 1OOA") to current_transaction
  endif
else

$INLINE
  printf("Gas Station Error: There is no instance of Pump number 2.\n");
$ENDINLINE

endif
```

```
# Threat Escapes from Platform 2 without paying

platform = find-one PLATFORM where Platform_Number = 2

if platform != UNDEFINED then

  current_pending_d3assessment = platform -> R9

  if current_pending_d3assessment != UNDEFINED then
    current_d3assessment = current_pending_d3assessment -> R4
    generate TRN3:Threat_Escapes("Unidentified enemy aircraft, signature indicates low observable fighter") to current_d3assessment
```

```
  endif
else

$INLINE
  printf("Air Warfare Error: There is no instance of Platform number 2.\n");
$ENDINLINE

endif
```

P.23.9 9 WEAPON LOGISTICS OFFICER Delivery For Weapons Store 1002

(Orig. TANKER Delivery For Tank 1002)

 # Tanker driver fills tank 1002 with 10000

tank = find-one TANK where Tank_Number = 1002

if tank != UNDEFINED then
generate TNK1:Filled_Tank(10000.00) to tank
else

$INLINE
printf("Gas Station Error: There is no instance of Tank number 1002.\n");
$ENDINLINE

endif

```
# Weapons Logistics Officer fills weapon store 1002 with 10

weapon_store = find-one WEAPON_STORE where Weapon_Store_Number = 1002

if weapon_store != UNDEFINED then
  generate WPS1:Logistics_Delivery(10) to weapon_store
else

$INLINE
  printf("Air Warfare Error: There is no instance of Weapon Store number 1002.\n");
$ENDINLINE

endif
```

Appendix Q

Academic Paper One: ICCRTS Conference

Q.1 Paper Abstract

[redacted]

[redacted]

(Johnson, Siemieniuch and Woodhead 2007a)

Q.2 Paper Reference

Johnson, P.W., Siemieniuch, C.E., Woodhead, M.A. (2007) Identifying and Assessing Appropriate System Architecture Options for Generic and Specific Mission Requirements. In: **12th ICCRTS Conference**, 19-21 June, 2007, Newport, Rhode Island, USA. Paper reference 203.

http://www.dodccrp.org/events/12th_ICCRTS/CD/html/papers/203.pdf

Appendix R
Academic Paper Two: SEAS DTC Conference

R.1 Paper Abstract

[redacted]

(Johnson, Siemieniuch and Woodhead 2007b)

R.2 Paper Reference

Johnson, P.W., Siemieniuch, C.E., Woodhead, M.A. (2007) A Method For Evaluating Appropriate System Architecture Options for Generic and Specific Mission Requirements. In**: 2nd Annual SEAS DTC Conference**, 10-11 July, 2007, Edinburgh, UK. Paper reference SER006.

http://www.seasdtc.com/events/2007_conference/papers/SER006.pdf

Appendix S

Academic Paper Three: IET Conference

S.1 Paper Abstract

[redacted]

(Johnson, Siemieniuch and Woodhead 2007c)

S.2 Paper Reference

Johnson, P.W., Siemieniuch, C.E., Woodhead, M.A. (2007) A Decision Making Support System utilising the Evaluation of Whole System Capability against Mission Requirements. In: **Institute of Engineering and Technology 2nd Autonomous Systems Conference**, 23rd November, 2007, London, UK. Paper reference 009, published conference CD:

ISBN 9780863418730

ISSN 0537-9989 Reference PEZ07 131

Appendix T

Academic Paper Four: SEAS DTC Conference

T.1 Paper Abstract

[redacted]

(Johnson, Siemieniuch and Woodhead 2008)

T.2 Paper Reference

Johnson, P.W., Siemieniuch, C.E., Woodhead, M.A. (2008) An Initial Set of Case Studies Utilising a Developed Decision Support System for Evaluating Available Systems Against Mission Requirements. In**: 3rd Annual SEAS DTC Conference**, 24-25 June, 2008, Edinburgh, UK. Paper reference SER006.

http://www.seasdtc.com/events/2008_conference/papers/SER006_paper.pdf