ARCHITECTING HUMAN OPERATOR TRUST IN AUTOMATION TO IMPROVE SYSTEM EFFECTIVENESS IN MULTIPLE UNMANNED AERIAL VEHICLE (UAV) CONTROL

Graduate Thesis

Eric A Cring
Captain, USAF

Adam G Lenfestey
Major, USAF

AFIT/GSE/ENV/09-M06

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION IS UNLIMITED

The views expressed in this thesis are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the United States Government.

AFIT/GSE/ENV/09-M06

ARCHITECTING HUMAN OPERATOR TRUST IN AUTOMATION TO IMPROVE SYSTEM EFFECTIVENESS IN MULTIPLE UNMANNED AERIAL VEHICLE (UAV) CONTROL

THESIS

Presented to the Faculty

Department of Engineering and Engineering Management

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Systems Engineering

Eric A Cring, BS, MBA
Captain, USAF

Adam G Lenfestey, BS
Major, USAF

March 2009

AFIT/GSE/ENV/09-M06

ARCHITECTING HUMAN OPERATOR TRUST IN AUTOMATION TO IMPROVE SYSTEM EFFECTIVENESS IN MULTIPLE UNMANNED AERIAL VEHICLE (UAV)

Eric A Cring, BS, MBA
Captain, USAF

Adam G Lenfestey, BS
Major, USAF

Approved:

John M. Colombi, Ph. D. (Chairman) Date

Vincent A. Schmidt, Ph. D. (Member) Date

Joseph W. Carl, Ph. D. (Member) Date

David S. Long, Lt Col (Member) Date

AFIT/GSE/ENV/09-M06

Abstract

Current Unmanned Aerial System (UAS) designs require multiple operators for each vehicle, partly due to imperfect automation matched with the complex operational environment. This study examines the effectiveness of future UAS automation by explicitly addressing the human/machine trust relationship during system architecting. A pedigreed engineering model of trust between human and machine was developed and applied to a laboratory-developed micro-UAS for Special Operations. This unprecedented investigation answered three primary questions. Can previous research be used to create a useful trust model for systems engineering? How can trust be considered explicitly within the DoD Architecture Framework? Can the utility of architecting trust be demonstrated on a given UAS architecture? By addressing operator trust explicitly during architecture development, system designers can incorporate more effective automation. The results provide the Systems Engineering community a new modeling technique for early human systems integration.

Acknowledgements

The authors would like to thank our advisors, Dr. John Colombi and Dr. Vince Schmidt, for their patience and insight during the last three terms; our committee members, Dr. Joseph Carl and Lt Col David Long, whose keen observations made this thesis a better product; and Ms. Lynn Curtis for her help in preparing this document for publication. We would also like to extend our appreciation to our research sponsor Dr. Janet Miller from the Air Force Research Labs, and her colleague Dr. Cheryl Batchelor, for sharing their professional expertise in human cognition and trust.

In addition, Mr. Mike Mueller and Mrs. Julie Naga provided valuable insight in Human Systems Integration, while Drs. Mike Patzek, Mark Draper, and Tom Hughes lent crucial experience gained while developing future control stations for single-operator control of multiple UAVs. Drs. Brian Tsou, Lamar Warfield , Justin Estepp and Benjamin Knott, meanwhile, contributed to our understanding of human performance measurement.

Last but not least, we extend special thanks to members of the 11th Reconnaissance Squadron at Creech AFB, NV, and the 1st Reconnaissance Squadron at Beale AFB, CA. In particular, we owe a debt of gratitude to Major Aaron Lattig for sharing his expertise in the operation of the PREDATOR UAV, and to Captains Fred Dohnke, Andy Schwaderer, Jeff Souza, and Brian Wells for providing their insights into GLOBAL HAWK operations. Their shared experiences and impressions of their weapon systems' automation provided a sanity check for many of the conclusions reached during the course of our research.

Adam G Lenfestey, Eric A Cring

Table Of Contents

Page

Abstract iv

Acknowledgements v

Chapter 1: Introduction 1

1.1 Motivation 1

1.2 Problem Statement 2

1.3 Objectives 5

1.4 Scope and Assumptions 5

1.5 Thesis Overview 6

Chapter 2: Literature Review 7

2.1 Overview and Scope 7

2.2 Trust and its Parameters 8

2.2.1 Trust Dependencies 9

2.2.1.1 Environment and Context 9

2.2.1.2 Actions Taken 10

2.2.1.3 Outcomes and Consequences 10

2.2.1.4 Information about Trustee 11

2.2.1.5 Operator Perception 11

2.2.2 Trust Bases / Expectations 16

2.2.2.1 Technical Competence / Ability 18

2.2.2.2 Persistence / Integrity 18

2.2.2.3 Fiduciary Responsibility / Benevolence 18

2.2.3 Trust Dynamics / Influences 19

2.2.3.1 Predictability 19

2.2.3.2 Dependability 20

2.2.3.3 Faith 20

2.2.4 Muir’s Model 21

2.3 Reliance 22

2.3.1 Misuse 23

2.3.2 Disuse 24

2.3.3 Reliance Mismatch Factors 25

Page

2.4 Design Recommendations for Incorporating Trust 26

2.4.1 Use Adaptive Automation 26

2.4.2 Display System Confidence 28

2.5 Train for Appropriate Trust 30

Chapter 3: Methodology 33

3.1 Macro View of Trust Relationships 34

3.2 Development of the SE Model of Trust 35

3.2.1 Perception 36

3.2.2 Context 37

3.2.3 Interaction of perception and context 38

3.2.4 Trust 40

3.2.5 Reliance Decision and its Outcome 41

3.2.6 Integrated Model of Trust 42

3.3 DoD Architecture Framework 43

3.3.1 Application of Trust Model to System Architecture 43

3.3.2 DoDAF v1.5 44

3.3.3 DoDAF v2.0 47

3.4 Zall/O’Malley Model 49

3.4.1 Architecting Cognition 49

3.4.2 Fundamental CPC Tasks 50

3.5 Architecting for Appropriate Trust 51

Chapter 4: Process for Integrating Trust Within DoDAF 52

4.1 The DoDAF Architecture Process 52

4.2 Architecture Methodologies 53

4.3 Architecting Trust 54

4.3.1 Determine Scope and View 55

4.3.2 Determine and Organize the Data 56

4.3.2.1 Functional / Behavioral 57

4.3.2.2 Informational 58

4.3.2.3 Sequence / Temporal 59

Page

4.3.2.4 Structural 60

4.3.3 Analyze and Document Results 60

4.4 Practical Application of the Trust Lens 61

4.4.1 Intro to BATCAM/CUSS Architecture 61

4.4.1.1 System Overview 62

4.4.1.2 Autopilot 63

4.4.1.3 Ground Station 63

4.4.1.4 Operations and Employment Concept 64

4.4.1.4.1 Surveil a Stationary Target BATCAM Operational Process 65

4.4.1.4.2 Provide Surveillance of a Series of Waypoints BATCAM Operational Process ... 65

4.4.2 Architecting Trust: Determining BATCAM Purpose and Scope 66

4.4.3 Trust Data Required to Support BATCAM Architecture Development 67

4.4.3.1 Functional/Behavioral 67

4.4.3.2 Informational 75

4.4.3.3 Sequence/Temporal 77

4.4.3.4 Structural 80

Chapter 5: Conclusions and Recommendations 82

5.1 Conclusions 82

5.1.1 Can previous research be used to create a useful trust model for systems engineering? 82

5.1.2 How can trust be considered explicitly within DoDAF? 84

5.1.3 Can the utility of architecting trust be demonstrated on a given UAS architecture? ... 85

5.2 Recommendations 86

5.2.1 Develop Comprehensive Test Plan to Quantify Utility of Architecting Trust 86

5.2.2 Extend Trust Model to Show Team Collaboration 86

5.2.3 Perform HFE Study to Determine Most Effective Means to Convey Trust Data 87

5.2.4 Develop State Transition Diagram for Evolution of Human/Machine Trust 87

5.2.5 Determine Appropriate Degree of Transparency 87

5.2.6 Develop a Trust Profile 88

Works Cited 90

Page

Appendix A – Trust Lens Implementation Guide.. 94
Appendix B – Glossary.. 95

List of Figures

Page

Figure 1 – Realistic Trust Enables Maximal Effectiveness 4
Figure 2 – Operators Verify Less as Time Stress Increases (Cohen, Parasuraman, & Freeman, 1998) 13
Figure 3 - Human Awareness Changes With Levels of Automation (Drury & Scott, 2008) 15
Figure 4 - Muir's Model of Trust (1994) 22
Figure 5 - Effects of Static and Adaptive Automation on Change Detection, SA, and Workload (Parasuraman, Barnes and Cosenzo, 2007) 27
Figure 6 - Hierarchical Control Loops (Cummings, Bruni, Mercier, & Mitchell, 2007) 28
Figure 7 - Antifakos, Kern, Schiele & Schwaninger, 2005 30
Figure 8 - Macro view of trust relationships 34
Figure 9 - UML Class Hierarchy of Operator Perception 36
Figure 10 - UML Class Hierarchy of Context 37
Figure 11 - Interaction of Perception, Context, and Trust 39
Figure 12 - Interaction of trust, perception, context and expectations 40
Figure 13 - Reliance Decision Tree 41
Figure 14 - SE Model of Trust Relationships 43
Figure 15 - Fundamental Linkages Among the Views, DoDAF v1.5 Vol. 2 45
Figure 16 - Notional Mapping of DoDAF v1.5 Views to Draft DoDAF 2.0 Views 48
Figure 17 - BATCAM UAV (Parsch, 2005) 62
Figure 18 - Autopilot System (Procerus Technologies, 2006) 63
Figure 19 - Ground Control Unit (Sakryd & Ericson, 2008) 64
Figure 20 - Portable Computing Device (Sakryd & Ericson, 2008) 64
Figure 21: BATCAM Operational Concept 66
Figure 22 - Provide Surveillance OV-5 Diagram 68
Figure 23: OV-5 Diagram with Trust Factors Added 69
Figure 24 - Manage UAVs OV-5 Activity Diagram 70
Figure 25: Updated OV-5 Diagram with Trust Factors 71
Figure 26 - BATCAM SV-4 Functional Decomposition Diagram 72
Figure 27: SV-4 Diagram with Trust Factors Added 73
Figure 28: SV-4 Diagram with Trust Factors Added 74
Figure 29: CUSS Related OV-7 Logical Data Model 75
Figure 30: CUSS Related OV-7 Logical Data Model with Trust Factors Included 77
Figure 31: OV-6c Operational Event Diagram for BATCAM Route Surveillance 78
Figure 32: Operational Event Diagram for BATCAM Route Surveillance with Trust Factors Added 79
Figure 33 - BATCAM SV-01 System Interface Diagram 80

List of Tables

Page

Table 1 - List of Views in DoDAF v1.5 46
Table 2 - Early Conceptual Design Eight Fundamental CPC Tasks (Zall & O'Malley, 2008) 50

Architecting Human Operator Trust in Automation to Improve System Effectiveness in Multiple Unmanned Aerial Vehicle (UAV) Control

Chapter 1: Introduction

1.1 Motivation

Unmanned Aerial Systems (UAS) have attained a prominent role in Joint warfighting over the last ten years, particularly in Operations ENDURING FREEDOM and IRAQI FREEDOM. While the air vehicles themselves are uninhabited, it is a fallacy to believe that humans are a less important component of the overall UAS than of any other aerial systems. In fact, some current UAS designs require multiple operators for each vehicle, largely due to the limited ability of existing automation to respond effectively to the complexity of the operational environment.

On December 6, 1999, a U.S. Air Force Global Hawk Unmanned Aerial Vehicle (UAV) ran off a runway at Edwards Air Force Base causing extensive damage to the vehicle. One of the contributing factors to the crash was overreliance on the system's automation. The UAV experienced an in-flight problem with the onboard temperature regulation system within the avionics compartment. The aircraft was commanded to return to base and the system's automation implemented a secondary contingency plan to conduct an automated landing and taxi. The preprogrammed automation instructed the aircraft to accelerate to a speed of 155 knots between any two waypoints if the altitude between the points varied by more than a pre-specified amount. Unbeknown to the operators, two of the waypoints on the taxi path exceeded the threshold and the aircraft accelerated to a speed where it was unable to negotiate a turn and ran off the runway (Williams, 2006).

Human operators under-relying on the Global Hawk UAV automation is another problem that is currently plaguing the platform, negatively impacting the effectiveness of the system. Based on crew interviews conducted in September 2008 at Beale Air Force Base, CA, current training emphasizes the need for operators to check the dialogue box on the system display after each command is given in order to ensure that the air vehicle is functioning as commanded. However, there is no confirmation from the air vehicle that a command was received until the aircraft is observed to execute the command. This lack of feedback from the automation or knowledge of its state causes operators to double check their work, which takes time and increases the cognitive workload of the operator. In addition, the operator's focus is diverted from other critical tasks during this period in order to develop sufficient situational awareness of system behavior to determine whether intervention is required (Cummings & Mitchell, 2008).

These two cases exemplify the fact that human operators have a tendency to either over rely or under rely on a system's automation. Underutilizing the automation increases the cognitive workload of the operator, which can adversely affect their ability to perform effectively. Over relying on the automation can have drastic consequences, including damage to or loss of the air vehicle. Therefore, it is essential that future systems are designed to provide the appropriate level and type of automation to facilitate collaboration between the human and the machine counterpart. In accomplishing this task, emphasis should be placed on the interaction between the human and machine, in order to create an environment that fosters an appropriate level of trust by the human operator(s), as will be discussed below.

1.2 Problem Statement

As the demand for UAS capability has grown, the Air Force's ability to produce mission-ready operators for these specialized systems has been stressed. This highlights one of the primary emphases within the Human Systems Integration (HSI) community. The community is accentuating the need to account for the human as *part* of the system during *early* system design to make system operation more effective and efficient. Rather than many operators per vehicle, future systems should incorporate effective automation to allow a single operator to control multiple vehicles performing a variety of tasks, potentially in synchronization, in a fluid operational environment. The challenge is to have the appropriate number of people with respect to the cognitive tasks.

Automation is “the technology that actively selects data, transforms information, makes decisions, or controls processes” (Lee & See, 2004). However, a large body of research shows that *more* automation is not necessarily *better* automation (Parasuraman, Barnes, & Cosenzo, 2007). In order to be effective, the automation must be well-designed, reliable, and tailored to complement the capabilities of the human operator in varying supervisory roles (Cummings, Bruni, Mercier, & Mitchell, 2007) (Cummings & Mitchell, 2008).

The best automation, however, is ineffective if not used or used improperly. In this regard, the operator’s level of trust in the automation is a vital factor in determining its operational effectiveness. This form of trust is also known as social or "pragmatic" trust due to its analog in human cognition (Konstantinou, Liagkou, Spirakis, Stamatiou, & Yung, 2005), as opposed to “trust” in terms of confidentiality, integrity, and other security related aspects of automated systems. Pragmatic trust, which is the focus of this research effort, can be defined as the attitude that an agent will help achieve an individual’s goals in a situation characterized by uncertainty and vulnerability (Lee & See, 2004). If a human operator trusts automation inappropriately and the automation fails, effectiveness is reduced; sometimes catastrophically. Conversely, if the operator chooses not to trust the automation and instead controls the system manually, human error and inefficiency can reduce effectiveness. Realistic trust that reflects the true capabilities of the automation enables maximum effectiveness for a given system. The authors of this research developed the following diagram to portray such a relationship between operator trust and mission effectiveness.

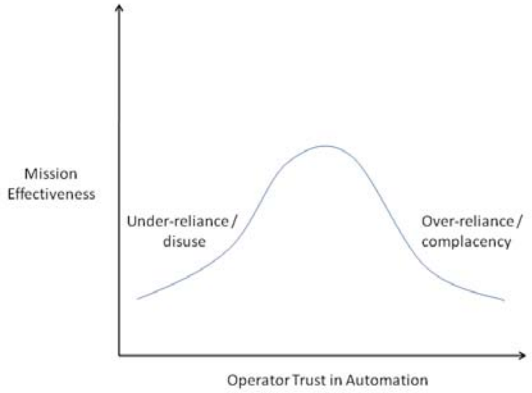


Figure 1 – Realistic Trust Enables Maximal Effectiveness

One problem is that trust, as with many aspects of human cognition, is difficult to quantify. If a robust metric existed for trust, systems engineers and program managers could use it to evaluate competing system designs. The lack of human performance metrics has been repeatedly noted in research on Human Systems Integration, which is "the interdisciplinary technical and management processes for integrating human considerations within and across all system elements" (INCOSE, August 2007). The community consensus is that better quantitative tools and methods are required in order to fully address this shortfall (Committee on Human-System Design Support for Changing Technology, 2007). Previous research efforts have identified common factors that influence trust (Antifakos, Kern, Schiele, & Schwaninger, 2005)**;** (Parasuraman, Sheridan, & Wickens, 2008)**;** (Dzindolet, Peterson, Pomranky, Pierce, & Beck, 2003), while others have proposed mathematical models of trust (Lee & See, 2004); (Bhattacharya, Devinney, & Pillutla, 1998); (Cohen, Parasuraman, & Freeman, 1998) and have attempted to measure trust empirically (Jian, Bisantz, & Drury, 2000). Building on this body of work, it may be possible now to improve overall mission effectiveness by explicitly designing systems to elicit appropriate levels of trust in UAS automation.

1.3 Objectives

This research will attempt to answer three main questions:

- Can previous research be used to create a useful trust model for systems engineering?
- How can trust be considered explicitly within the DoD Architecture Framework (DoDAF) in order to improve the effectiveness and suitability of future UAS designs?
- Can the utility of architecting trust be demonstrated on a given UAS architecture?

Results can then be used by capability planners and early DoD acquisition processes in order to address trust factors explicitly during design, thereby improving the effectiveness of automation in UAS architectures.

1.4 Scope and Assumptions

Trust has been the focus of myriad studies in numerous disciplines. This thesis cannot possibly encompass all previous work related to trust, nor address trust for all situations humans may encounter. Rather, it will focus on architecting trust in the context of effective UAS control. This effort will focus on architecting trust by adapting existing research to (1) determine how trust manifests in various aspects of UAS architecture, and (2) how each factor can be addressed to foster an appropriate level of operator trust in the system.

Several assertions must be made to effectively address trust within a system architecture. The first is that trust is not self-contained or binary, but rather is composed of a set of cognitive inputs that act in combination and along a continuum. The second is that, while there will be some variance between individuals, humans will tend to respond to those inputs in ways similar enough to predict. This predicted response will be especially true of military operators with common training, organization, and ethos. Lastly, while the fundamental physical mechanisms of human cognition are not completely understood, the effects of human cognition on decision-making can be quantified. These assertions are supported by decades of empirical evidence from trust-related research, which will be described in Chapter 2.

1.5 Thesis Overview

Chapter 2 will provide an overview of the extensive body of work related to trust in automation and will also describe the fundamentals of UAS automation with an emphasis on factors that affect operator trust. Chapter 3 will provide a detailed analysis of trust and its component elements through the development of a systems engineering model of trust. In addition, Chapter 3 will introduce the Department of Defense Architecture Framework (DoDAF) and related architecture development concepts.   Chapter 4 will depict the application of the trust lens to the architecture process and demonstrate its applicability to improve the architecture of an existing UAS. Chapter 5 will summarize our findings and identify recommendations for further research.

Chapter 2: Literature Review

2.1 Overview and Scope

As previously stated, trust relationships have been extensively researched across many disciplines, including psychology, human factors, ergonomics, management, and systems engineering. Much of the large body of research on the topic has dealt with human-human trust. Yet, as we interact with automation more and more frequently in our daily lives, the topic of human-machine trust has steadily gained importance. Human-machine trust is especially relevant in operations involving UAS, because human operators are almost entirely reliant on system feedback due to their physical separation from the air vehicle. What follows is a targeted summary of those works most directly relevant to the thesis objectives: identifying trust factors inherent in single-operator multi-UAV control and considering those factors explicitly when developing UAS architectures.

Research related to human performance and interaction with automation began as early as the 1960s (Senders, 1964); (Fitts & Posner, 1967) and began to focus on automation reliability and resulting human operator trust in the 1970s (Halpin, Johnson, & Thornberry, 1973). Subsequent research focused on identifying social and psychological factors that influenced trust (Barber, 1983), categorizing user acceptance or rejection of unfamiliar automation in industrial applications (Zuboff, 1988) and analyzing the trustworthiness of automation (Sheridan, 1988). More recently, researchers have attempted to identify and model the relationships of factors that influence operator trust, (Muir, 1994) (Bhattacharya, Devinney, & Pillutla, 1998) (Cohen, Parasuraman, & Freeman, 1998), to measure trust empirically (Jian, Bisantz, & Drury, 2000), and to predict operators' tendency to trust and rely on automation under various conditions (Bisantz & Seong, 2001) (Dzindolet, Pierce, Beck, Dawe, & Anderson, 2001) (Dzindolet, Peterson, Pomranky, Pierce, & Beck, 2003) (Gao & Lee, 2006). These and similar works defined the parameters of trust, such as the concepts of appropriate trust, misuse, disuse, reliability, and others, which will be reviewed shortly. Lee & See (2004) summarized these and roughly 200 other published works on trust, and proposed specific recommendations to enable designers to elicit appropriate operator reliance on automation. Lee & See's findings provide a framework for much of the remainder of this chapter.

Most recently, two studies have proposed automation design guidelines for a single operator/multiple UAV system (Cummings & Mitchell, 2008) (Cummings, Bruni, Mercier, & Mitchell, 2007). These studies did not address trust explicitly, but examined situational awareness and attention; therefore they present an opportunity to apply the concepts from this chapter.

2.2 Trust and its Parameters

The definition of trust varies depending on the context in which it is used, as has been illustrated by much of the early research on the subject. Bhattacharya *et al.* (1998) established several characteristics useful in defining trust: First, trust exists in an uncertain and risky environment, and is unnecessary when certainty exists. Second, trust represents an expectancy by the trustor that reflects some aspect of predictability. Third, any definition of trust must account for trust's strength and importance, where strength reflects the degree of confidence the trustor has in the expected outcome, and importance represents the value of the expected outcome to the trustor. Fourth, trust is situation and person specific. And last, trust reflects the degree of expectation of a positive outcome (Bhattacharya, Devinney, & Pillutla, 1998). In short, whether the trustee "agent" is automated or human trust has been defined as:

> **"the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability."** (Lee & See, 2004)

It is important to draw a distinction between trust and reliance. Trust is an attitude, and reliance is a behavior reflecting an intention or willingness to act. According to Lee & See (2004), “trust stands between beliefs about the characteristics of the automation and the intent to rely on the automation.” Further, “trust guides – but does not completely determine – reliance.”

2.2.1 Trust Dependencies

Trust and trustworthiness are not innate qualities, but rather depend on many factors. As mentioned previously, trust is situation and person specific, and often varies over time. Yet, common elements appear repeatedly in the literature regarding trust.

2.2.1.1 Environment and Context

The environment in which automation is being used plays an important role in an operator’s need to trust. The environment’s importance is highlighted by the Human Systems Integration (HSI) community that classifies environment as one of the nine primary HSI domains. As discussed by Bhattacharya, *et al.* (1998), trust is unnecessary when outcomes are certain. Conversely, trust is an important factor in highly uncertain and risky environments, where decisions must be made quickly with imperfect information. In such situations, an individual’s propensity to trust is shaped by individual, cultural, and organizational context.

Cultural and organizational contexts influence trust as individuals interact in ways that inform them about the trustworthiness of others and by establishing general norms and expected behaviors. Individual context refers to a specific person’s history of interactions with the trustee and their general inclination to trust, which can vary from one individual to the next.

Individual variation in tendency to trust is important to consider when evaluating automation, since variations may affect reliance in ways that are not directly related to the characteristics of the automation. While this must be taken into account by designers, previous research has shown that trust tendency as a personality trait can be reliably measured, and that the trust tendency influences behavior in predictable ways (Lee & See, 2004).

2.2.1.2 Actions Taken

Actors (be they human or machine) engage in actions intended to produce specific results and these individual decisions and actions act together to produce outcomes. However, since causality can be difficult to determine in an uncertain environment, it is not always possible to predict the consequences of alternative courses of action. In some situations, one individual will know the outcome of the other's action before being required to act, while in other situations each must act simultaneously or without knowledge of the consequences of each other's actions. In either case, individuals will predict the other's action and subjectively assign it a probability. This probability represents the *strength* of their expectation, or level of confidence in the trustee's predictability (Bhattacharya, Devinney, & Pillutla, 1998). High predictability is a major factor in fostering trust in automation, since it allows a trustor to establish accurate expectations of the automation's capability and behavior.

2.2.1.3 Outcomes and Consequences

"Clearly, trust concerns an expectancy or an attitude regarding the likelihood of favorable responses" (Lee & See, 2004). In addition to predicting a specific action, individuals will also assign a probability that the actions taken by the other will be appropriate and effective toward achieving the desired outcome. This prediction is influenced by the degree of *importance* the individual attaches to achieving a specific outcome, which is often related to the perceived consequences of success or failure. Achievement or failure to attain a favorable outcome affects future trust by creating, reinforcing, or refuting expectations of the trustee's competence (Bhattacharya, Devinney, & Pillutla, 1998). For example, the crash of a multimillion dollar UAV usually generates intense pressure to isolate and correct the cause. Until that is done, UAV operators may lose trust in the system. Interestingly, though, some studies have shown that high complacency individuals, or those with high levels of trust in automation, were actually more successful in making accurate predictions, and thus more likely to detect automation failures as they happened, than low-trust individuals (Lee & See, 2004).

2.2.1.4 Information about Trustee

Bhattacharya *et al.* (1995) describe trust as a multidimensional construct that is based on trustee characteristics. The relevant trustee characteristics are any that inform the individual about the ability of the trustee to achieve the trustor's goals. These include the nature, competence, and trustworthiness of the trustee, among other factors. One of the major questions is "what is to be trusted?" This is also known as specificity which is "the degree to which trust corresponds to a particular component or aspect of the trustee." For example, an operator may have different levels of trust for a system as a whole than for a specific mode of a given subsystem. (Lee & See, 2004)

2.2.1.5 Operator Perception

Operator perception can influence trust and affect reliance in many ways. Beliefs and perceptions are part of the information base that determines the attitudes that affect operator trust and reliance. Self-confidence, perceived mental workload, situational awareness, and time stress are the major perceptions that an operator will create that will directly influence his/her trust in the automation and reliance upon it.

Lee and See (2004) claim the level of trust combines with other attitudes and expectations, such as operator workload and self-confidence, to determine the intention to rely upon the automation of a system. If an operator has low trust in the system and high self-confidence, then they will tend to disuse the automation. Likewise, if an operator has high trust in system automation and their self-confidence is low, then they will tend to misuse the automation. In general, individuals tend to rely more upon automation when their confidence in their own ability is lower than in the automation, and vice-versa. These biases in self-confidence can have a substantial effect on the appropriate reliance on automation (Lee & See, 2004).

These results were demonstrated in a series of tests. A study of a vehicle navigation aid found that system errors cause trust and reliance to decline more for those in a familiar city, whose self-confidence was high, as compared with those in an unfamiliar city, whose self-confidence was low (Lee & See, 2004). Likewise, in a study between pilots and students, students were found to have more self-confidence and therefore were less inclined to use the given automation than were the pilots who were familiar with the automation and tended to rely upon it more (Lee & See, 2004).

Mental workload can be described as the relation between the function relating the mental resources of a task and those resources available to be supplied by the human operator (Parasuraman, Sheridan, & Wickens, 2008). Human mental workload is paramount in developing a successful transition from current operations of many operators to one UAV to one operator controlling multiple UAVs. The challenge in accomplishing this transition is to increase the level of system autonomy in an effective way that will reduce human workload (Cummings & Mitchell, 2008).

However, simply increasing system automation will not necessarily improve system performance through a reduction in mental workload. In abnormal or unexpected situations, the automation could fail; possibly causing a catastrophic event to occur while the operator may not be engaged in the task (Cummings & Mitchell, 2008). The lack of transparency and trust in the system can potentially lead to increased operator mental workload in attempting to determine whether or not the automation is working correctly and if intervention is required. Therefore, architects should consider operator trust when evaluating levels and types of automation in order to reduce mental workload, wait times, interactions times, and queuing times, while improving the effectiveness of the system.

Cohen *et al*. (1998) also stressed the importance and relationship of mental workload/time stress and trust in decision making. They proposed a direct trade-off exists with respect to time between trust and decision-making, as shown in Figure 2. If an operator's trust in the aid's conclusion falls in the upper region, then the user should simply accept the conclusion without taking further time. If the trust falls in the lower region, the user should reject the aid's conclusion without further delay. Finally, if trust falls in the intermediate region, then it is worthwhile for the user to take time to decide on a course of action.

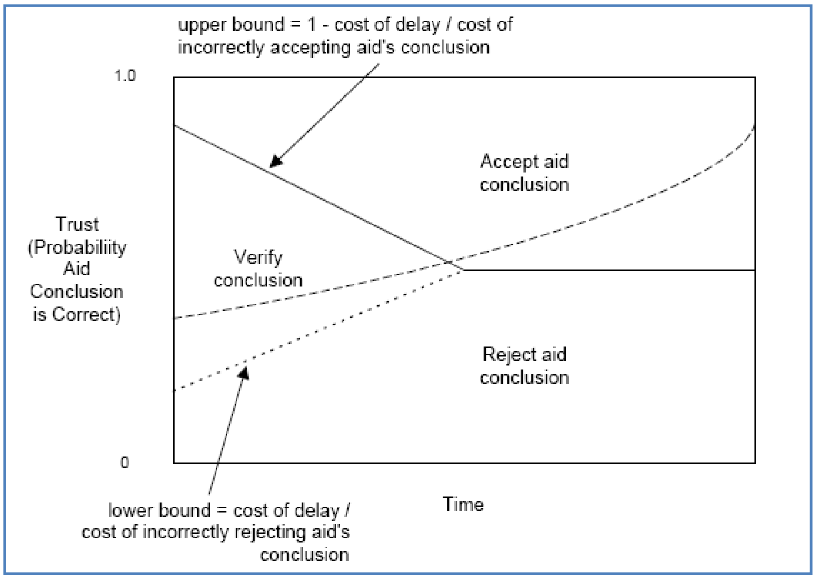
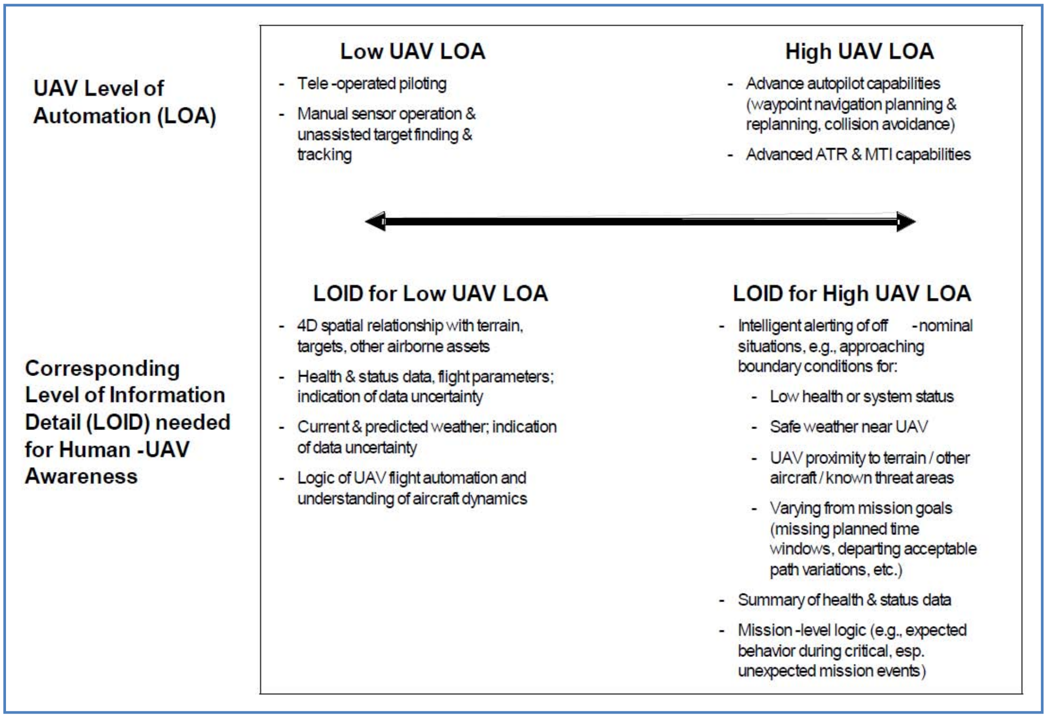

upper bound = 1 - cost of delay / cost of incorrectly accepting aid's conclusion
1.0
Accept aid conclusion
Trust (Probabiliity Aid Conclusion is Correct)
Verify conclusion
Reject aid conclusion
0
Time
lower bound = cost of delay / cost of incorrectly rejecting aid's conclusion

Figure 2 – Operators Verify Less as Time Stress Increases (Cohen, Parasuraman, & Freeman, 1998)

The upper dashed line in the figure portrays the typical amount of trust an operator has in a particular automation over time. In the beginning, the initial trust is typically low and depends upon factors such as context, predictability, and expectations. The relationships between these factors will be explained in detail later in the chapter. The user's confidence in the automation typically increases as more information is collected and the trust relationship matures (Cohen, Parasuraman, & Freeman, 1998).

The key variables that influence the reliance decisions in the model above are uncertainty and time stress. Time stress acts as the main cost of delay parameter in determining the upper and lower bounds on the figure. As the Figure 2 depicts, action is more imperative when the cost of delay is great, even with high uncertainty about trust. In addition, as time stress increases the upper and lower bounds move towards each other, indicating the need to make a decision with less verification at the same levels of trust (Cohen, Parasuraman, & Freeman, 1998).

Situational awareness (SA) is "the perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future" (Endsley & Garland, 2000). SA is not a choice or action taken based upon presented information, or a consequence of the diagnosis, nor is it a measure of a human operator's performance. Instead, SA is the operator's mental model of the current and near-future situation. Studies have shown that SA varies inversely as the level of automation is increased and decreased in UAV operations. Drury & Scott (2008) developed the following diagram, Figure 3, to portray this concept.



UAV Level of Automation (LOA)

Low UAV LOA

- Tele -operated piloting
- Manual sensor operation & unassisted target finding & tracking

High UAV LOA

- Advance autopilot capabilities (waypoint navigation planning & replanning, collision avoidance)
- Advanced ATR & MTI capabilities

Corresponding Level of Information Detail (LOID) needed for Human -UAV Awareness

LOID for Low UAV LOA

- 4D spatial relationship with terrain, targets, other airborne assets
- Health & status data, flight parameters; indication of data uncertainty
- Current & predicted weather; indication of data uncertainty
- Logic of UAV flight automation and understanding of aircraft dynamics

LOID for High UAV LOA

- Intelligent alerting of off -nominal situations, e.g., approaching boundary conditions for:
  - Low health or system status
  - Safe weather near UAV
  - UAV proximity to terrain / other aircraft / known threat areas
  - Varying from mission goals (missing planned time windows, departing acceptable path variations, etc.)
- Summary of health & status data
- Mission -level logic (e.g., expected behavior during critical, esp. unexpected mission events)

Figure 3 - Human Awareness Changes With Levels of Automation (Drury & Scott, 2008)

As depicted, when the level of automation of a UAV increases, the level of information detail (LOID) and thus operator’s associated SA with regard to the UAV will tend to decrease. For example, the Global Hawk UAV is a highly automated air vehicle that has the ability to fly between pre-programmed way points, and therefore human operators need less SA of the aircraft’s flight-related information than operators would need of a low-automation Predator UAV pursuing a time sensitive target (Drury & Scott, 2008).

Past research efforts have categorized SA into the following three levels: level 1 – perception of information, level 2 – comprehension of current situation, and level 3 – projection of future status. Parasuraman *et al*. (2008) stress the notion that these levels can be accurately measured to provide an understanding of the SA that the current system automation is providing.

For example, visual scanning tools are commercially available to aid in the assessment of level 1 SA (Parasuraman, Sheridan, & Wickens, 2008).

SA will also vary depending upon other factors such as an operator's trust in the automation. As mentioned earlier, as a system becomes more reliable, operators tend to develop more trust in the automation and have a tendency to develop an automation bias towards it. Previous research efforts have pointed to a correlation between automation bias and decreased SA. Likewise, a lack of trust in automation tends to foster an environment where operators closely supervise the system, thus potentially increasing their SA with regards to the UAV operations (Cummings & Mitchell, 2008). This illustrates a complex relationship that varies over time between perceptions like SA and mental workload, operator trust and its effect on reliance, and the resulting impact on mission effectiveness.

2.2.2 Trust Bases / Expectations

The diverse and copious research that has been conducted in the area of trust has generated numerous definitions. Many of these definitions focus on trust as an attitude or expectation that a human operator has towards the automation. The group of researchers in this category have defined trust in some of the following ways: "expectation related to subjective probability an individual assigns to the occurrence to some set of future events", "expectation of technically competent role performance"; or "expectations of fiduciary obligation and responsibility, that is, the expectation that some others in our social relationships have moral obligations and responsibility to demonstrate a special concern for others' interests above their own" (Lee & See, 2004).

Lee and See (2004) claim that trust evolves in a complex individual, cultural, and organizational context. Within this context the role of expectations plays a vital part. For example, they state that the individual follows a social learning approach in which expectations for a particular situation are determined by specific previous experiences with situations that are perceived as similar. They also report that trust is a generalized expectancy that is independent of specific experiences, but rather based on the generalization of a large number of diverse experiences (Lee & See, 2004). In other words, the prior experiences an individual brings into a trust relationship help to shape his/her expectations of the trustee, and those expectations are likely to affect the operator's future interaction with the trustee. This illustrates the iterative and evolving nature of trust.

Likewise, Lee and See (2004) state that the organizational context reflects the interactions between people that inform them about the trustworthiness of others, which can include reputation and gossip, while the social context is comprised of a set of norms and expectations. Therefore, individuals enter into arrangements with each other, or with sources of automation, with a predetermined level of trust based upon these contexts. For example, Lee and See (2004) assert that people will initially trust an engineer not because of the ability of any specific person, but because of the underlying education and regulatory structure that governs people in the role of an engineer. In other words, they trust because of an expectation that an engineer has earned some measure of trust by virtue of professional competence.

Bonnie Muir (1994), meanwhile, states that a human operator's decision to allow (or conversely override) automation to control a process demonstrates a certain level of trust in the automation. Therefore, it is necessary to have an appropriate model of trust in human-machine interaction to provide a basis for developing effective automation. The model presented in this research assumes a definition of trust consistent with Lee and See and expands it by describing the multidimensional character of trust, including three specific bases, or expectations: technical competence, persistence, and fiduciary responsibility (Barber, 1983). These bases appear similar in concept to those described by Lee & See (2004) as ability, integrity, and benevolence.

2.2.2.1 Technical Competence / Ability

Technical competence simply means that the trustor believes the trustee has the ability to perform the assigned task successfully (Muir, 1994). Lee & See (2004) define ability as "the group of skills, competencies, and characteristics that enable the trustee to influence the domain." The trustor's perception of the trustee's ability, as well as similar characteristics such as reliability and predictability, contribute to some expectation of performance, which is a primary component of trust especially in the early stage of a trust relationship. Section 2.2.3 will discuss performance's relationship to trust.

2.2.2.2 Persistence / Integrity

Persistence refers to an expectation of constancy, which, for example, allows the operator to construct a mental model of the automation through experience interacting with it. (Muir, 1994) This is similar to integrity, which Lee & See (2004) define as "the degree to which the trustee adheres to a set of principles the trustor finds acceptable." A trustor's perception of the trustee's integrity can be considered to derive from observations of ability over a prolonged period, and affords the trustor an expectation of dependability, which is another primary dynamic of trust, described in section 2.2.3.

2.2.2.3 Fiduciary Responsibility / Benevolence

Finally, fiduciary responsibility applies when trustee competence is unknown, and the trustor must assume the trustee will act appropriately. This often occurs when an operator assumes a system will meet its design-based performance criteria when it is operating as intended (Muir, Trust in Automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems, 1994). This is similar to Lee and See's (2004) term benevolence, which they define as “the extent to which the intents and motivations of the trustee are aligned with those of the trustor.” In a mature trust relationship, the trustor’s perception of trustee benevolence prompts faith, which is a judgment that the trustee can be relied on, even though specific actions and their outcomes may be unknown at the time.

2.2.3 Trust Dynamics / Influences

In addition to these three bases (persistence/integrity, competence/ability, and fiduciary responsibility/benevolence); Muir identifies three dynamics of trust: predictability, dependability, and faith (Rempel, Holmes, & Zanna, 1985). These dynamics appear consistent with Lee and See’s (2004) factors of performance, process, and purpose. Each of these dynamics applies to greater or lesser degrees depending on the maturity of the trust relationship, and each requires varying amounts and types of information in order to sustain appropriate trust.

2.2.3.1 Predictability

Predictability is important in the early stages of a trust relationship, and requires visibility of specific behaviors, or performance. For example, operators will judge automation based on its ability to deliver consistent and desirable results. However, several factors influence predictability judgments. First is actual predictability: the automation’s performance. Lee & See (2004) describe performance as “*what* the automation does. More specifically, performance refers to the competency or expertise as demonstrated by its ability to achieve the operator’s goals.” Next is perceived predictability, or transparency. The operator must be able to observe the automation’s behavior in order to estimate predictability. In general, simple automation will be more predictable and observable than automation that has many degrees of freedom. Lastly, the stability of the environment may affect predictability, since an unstable environment may cause difficulty for an operator to distinguish apparent unpredictability, which should be ignored, from inherent unpredictability, which should reduce trust (Muir, 1994). During developmental and operational testing, testers carefully measure the performance of system components and conduct repeated, controlled trials to isolate all possible variables, in order to minimize apparent unpredictability. However, exercising all variables is not possible in the operational environment. Operators must rely on built-in transparency, realistic training, and prior experience to accurately predict system behavior.

2.2.3.2 Dependability

As the trust relationship matures, focus changes from specific behaviors to an assessment of general dependability, or the extent to which the trustee can be relied upon. Operators who are familiar with the automation will begin to trust based on overall experience rather than specific behaviors, and will have a greater understanding of the situations in which the automation may be untrustworthy. Lee & See (2004) describe this tendency in terms of process, which is "the degree to which the automation's algorithms are appropriate for the situation and able to achieve the operator's goals." Operator judgment of automation dependability is enhanced when operators push the boundaries of the automation's capabilities into uncertain and risky scenarios, allowing them to observe how the automation reacts beyond its design limits (Muir, 1994). This "pushing the envelope" often happens during test & evaluation or training, but may also occur in the operational environment. In sum, Lee & See (2004) assert that "operators will trust the automation if its algorithms can be understood and seem capable of achieving the operator's goals in the current situation." This assertion is reinforced by inferences the operator draws by observing the automation's performance over a period of time.

2.2.3.3 Faith

In the final stage, faith represents a fully mature trust relationship. Here, faith is defined as "a closure against doubt in the face of an uncertain future."   In this stage, the operator's perception of the appropriateness and flexibility of the automation allows him to control the system effectively, even without a complete understanding of the automation’s complex behavior. Faith, in this context, is similar to purpose, which Lee & See (2004) define as “the degree to which the automation is being used within the realm of the designer’s intent.” This assumes the designer’s intent has been communicated to the operator. However, the operator also realizes the potential for unforeseen interactions and the limitations inherent in any mitigating procedures (Muir, 1994).

2.2.4 Muir’s Model

The integrated model of the relationship between automation, the operator’s trust, and predictions about the automation’s behavior developed by Muir (1994) is depicted in Figure 4. This model depicts the multidimensional trust construct. As previously discussed, human operator’s expectation of the automation’s persistence, competence, and responsibility plays an important role in establishing the initial level of trust in the automation. In addition, system and operator confidence is related to the plausibility of the inferences made. In this context, confidence refers to a level of expectation associated with a particular prediction, rather than the overall trust relationship. Muir's model allows a qualitative assessment of the appropriateness of operator trust in a given system or component (Muir, 1994).

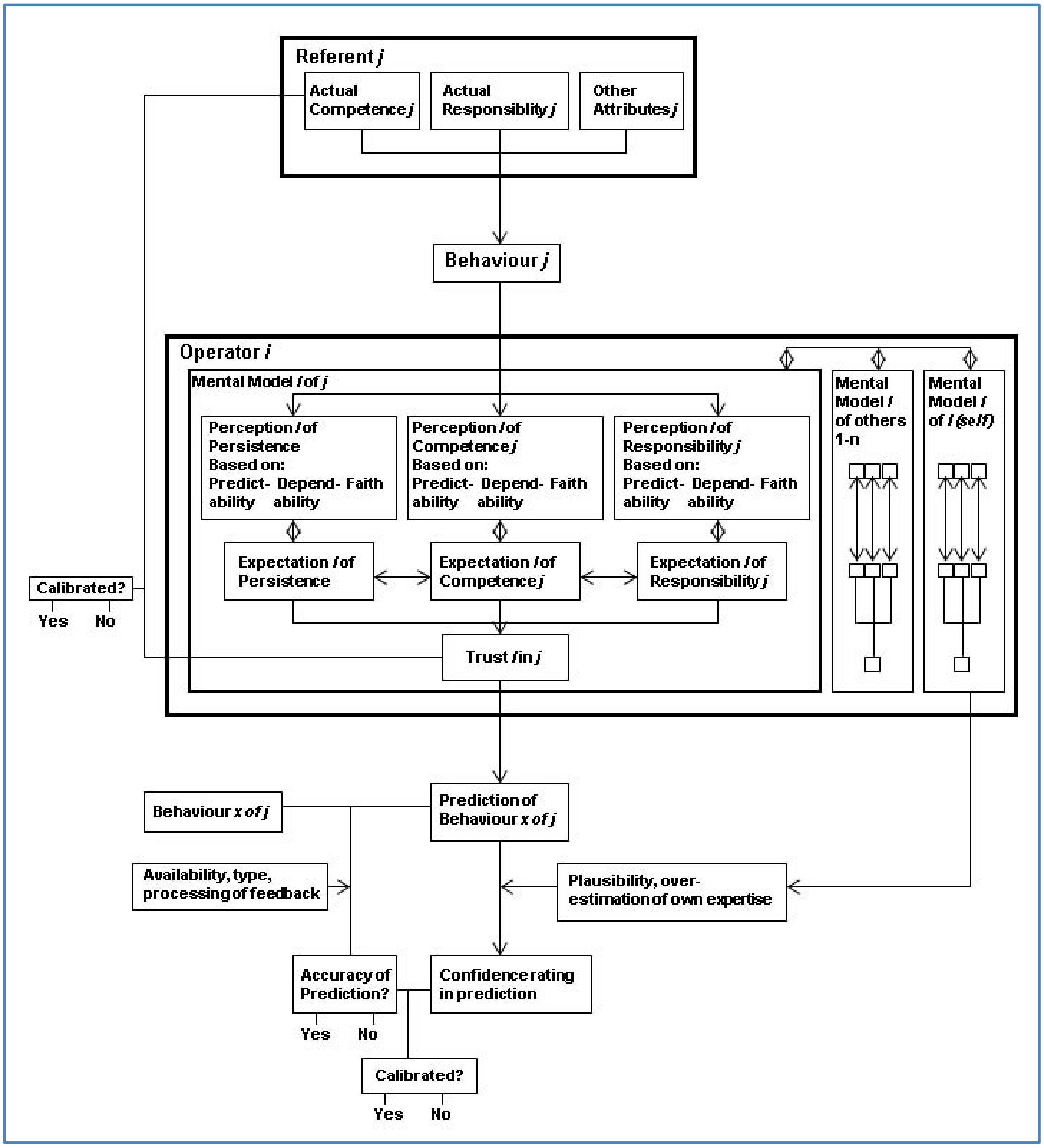

Referent j
Actual Competence j
Actual Responsiblity j
Other Attributes j
Behaviour j
Operator i
Mental Model I of j
Perception I of Persistence Based on: Predict- ability Depend- ability Faith
Perception I of Competence j Based on: Predict- ability Depend- ability Faith
Perception I of Responsibility j Based on: Predict- ability Depend- ability Faith
Expectation I of Persistence
Expectation I of Competence j
Expectation I of Responsibility j
Mental Model I of others 1-n
Mental Model I of I (self)
Calibrated?
Yes
No
Trust I in j
Behaviour x of j
Prediction of Behaviour x of j
Availability, type, processing of feedback
Plausibility, over-estimation of own expertise
Accuracy of Prediction?
Yes
No
Confidence rating in prediction
Calibrated?
Yes
No

Figure 4 - Muir's Model of Trust (1994)

2.3 Reliance

Many studies have demonstrated that in terms of man-machine interaction, "trust is an attitude toward automation that affects reliance and … can be measured consistently" (Lee & See, 2004). While others have characterized trust as an intention or willingness to act, Lee & See (2004) assert that trust is an attitude that affects reliance. In this context, reliance is the operator's intention or willingness to act, in the form of a decision to use or not to use automation. While trust is not the only influence on operator reliance, it is a key factor "when the complexity of the automation makes a complete understanding impractical and when the situation demands adaptive behavior that procedures cannot guide" (Lee & See, 2004). These conditions often result in inappropriate reliance, which falls generally into two categories: misuse and disuse (Parasuraman & Riley, 1997). Misuse refers to failures that occur when operators rely on automation in excess of its capabilities, i.e. over-trust. Disuse refers to operator rejection of automation when it is capable, i.e. under-trust.

2.3.1 Misuse

Human operators will tend to rely more upon automation as its reliability and their understanding of the automation increases. In some cases, this can lead to overreliance, or *automation bias*. Automation bias is a decision bias that occurs when operators become over-reliant on the automation and do not verify the accuracy of automated recommendations. This situation could lead to complacency and erroneous, if not catastrophic, errors (Cummings, Bruni, Mercier, & Mitchell, 2007).

Very high levels of trust in automation that are not perfectly reliable can be associated with overreliance and failure to monitor the "raw" information sources that provide input to the automated system. This phenomenon is known as complacency (Parasuraman, Sheridan, & Wickens, 2008). For example, if operators are given a tool to measure these "raw" information sources, complacency can be measured by evaluating whether or not the operator uses the tool more or less often under automation than under manual control. Complacency, as reflected by a strategy of allocating attention away from the automated task to other concurrent tasks, can be measured using eye recording devices. Using this approach, the eye movement recordings should show that operators scan the raw information sources less frequently when using automation than when performing the task manually or less frequently when automation reliability is higher than lower (Parasuraman, Sheridan, & Wickens, 2008).

This concept was demonstrated in a study by Senders (1964). In his research, he showed that human observers tasked with monitoring for abnormal readings on multiple dials with different frequencies of changing values made eye movements to the dials in proportion to bandwidth (Senders, 1964). Moray and Inagaki expanded on this study suggesting that a human operator who monitored automation at a rate less than the optimal Nyquist frequency (which represents a sampling rate twice that of the highest frequency present in the sampled data) was complacent while the one who monitored at a greater rate was skeptical. Individuals monitoring at the Nyquist rate are considered to be well calibrated (Moray & Inagaki, 2000). This approach may provide system engineers a proxy measure for trust; however, its applicability in design still needs to be proven.

2.3.2 Disuse

Past studies of human-machine interaction have shown that under-reliance, or disuse, tends to be less common than misuse in many situations (Dzindolet, Pierce, Beck, Dawe, & Anderson, 2001). When disuse does occur, it reduces the benefits technology offers and can negatively impact mission effectiveness (Pina, Donmez, & Cummings, 2008**.** Disuse is most commonly caused by false alarms, which reduce operator trust over time and increase workload (Parasuraman & Riley, 1997). Disuse due to false alarms can be decreased through training, specifically by providing the operator knowledge of the automation's error rate (Dzindolet, Pierce, Beck, Dawe, & Anderson, 2001). However, not all disuse is trust-induced. Design and ease of use are also factors, as in cases where automation requires substantial time to configure and activate. An operator under time constraint or high workload may choose not to rely on automation for these reasons, even though the automation is highly trusted to produce effective results (Lee & See, 2004). For example, a pilot concentrating on landing an aircraft may be inclined to disuse its flight management system if air traffic control changes require the automation to be reprogrammed during the last minutes of final descent. (Parasuraman, Barnes, & Cosenzo, 2007)

2.3.3 Reliance Mismatch Factors

Appropriate reliance exists when the operator’s level of trust in the automation matches the automation’s true capabilities, which may not be clearly or accurately understood by the operator. (Improving this relationship is a key goal of this research.) Mismatches between operator trust and automation capability can be described in terms of three factors: calibration, resolution, and specificity (Lee & See, 2004). Calibration is the direct correlation between the operator’s trust and the automation’s capability, similar to misuse/disuse (Lee & Moray, 1994); (Muir, 1987). Resolution is the degree to which changes in automation capability affect operator trust (Cohen, Parasuraman, & Freeman, 1998). For example, with low resolution, a large change in capability will have little effect on trust. Lastly, specificity refers to the degree to which trust correlates to a specific component or aspect of the automation. Specificity can be further differentiated as functional, in which trust is modulated by specific functions, sub-functions, and modes of operation, and temporal, in which trust is modulated by changes in context that affect the automation’s capability (Lee & See, 2004).

The objective is for the operator to have good calibration, high resolution, and high specificity in order to achieve appropriate trust. Lee & See (2004) contend that these principles should “guide design, evaluation, and training to enhance human-automation partnerships.”

2.4 Design Recommendations for Incorporating Trust

2.4.1 Use Adaptive Automation

Cummings *et al*. (2007) present different levels of automation based on those developed by Parasuraman *et al.* (2000) in order to conduct a trade study to determine the benefits and shortfalls of each in regard for single operator, multiple UAV command and control. One of the levels of automation they present is *management-by-exception.* This level occurs when automation decides to take an action based on some set of pre-determined criteria, and only gives operators a chance to veto the automation's decision (Cummings, Bruni, Mercier, & Mitchell, 2007). Their findings portray management-by-exception as dangerous due to its tendency to foster an automation bias atmosphere.

This condition of automation bias, or overreliance, was one of the critical contributing factors that led to fratricide during Operation Iraqi Freedom. In 2004, a U.S. Army Patriot System engaged and shot down a British Tornado and an American F/A-18. The system was designed to use a management-by-exception automation where operators were given 15 seconds to veto an automation decision. In this case, the system was highly complex, confusing, and comprised of often incorrect displays (Cummings, Bruni, Mercier, & Mitchell, 2007). This situation led the operators to become over reliant upon the automation and they failed to veto the computer's decision, which resulted in three coalition deaths.

If, instead, the automation adapts to the needs of the operator in a given situation, mission effectiveness can be enhanced by making the best use of human-machine collaboration. Parasuraman, *et al.* (2000) proposed a four-stage model of human information processing, composed of information acquisition, information analysis, decision and action selection, and action implementation.[1] They then proposed that these stages could benefit from different levels of automation, but that the specific implementation would be situation and application dependent (Parasuraman, Sheridan, & Wickens, 2000). For example, Parasuraman *et al*. (2007) simulated a combat identification system in which operators were asked to identify changes to a situation map. They demonstrated that adaptive automation, which activated when operator performance fell below a threshold, improved SA and change detection while lowering mental workload. Figure 5 depicts the results of the simulation (Parasuraman, Barnes, & Cosenzo, 2007).

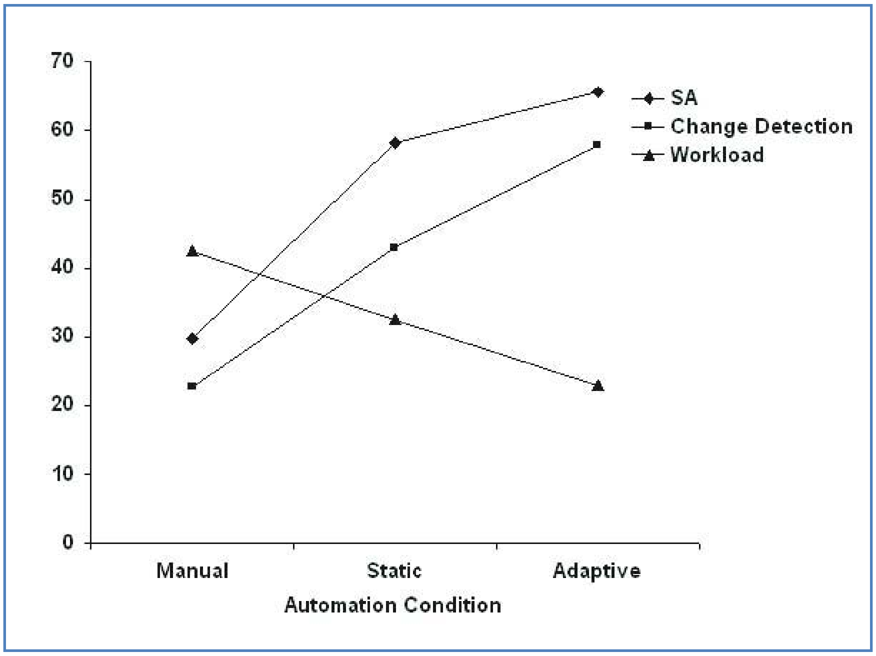


Figure 5 - Effects of Static and Adaptive Automation on Change Detection, SA, and Workload (Parasuraman, Barnes and Cosenzo, 2007)

Cummings, *et al.* (2007) extended this concept to single operator, multi-UAV control. They present a series of hierarchical control loops as shown in Figure 6, ranging from overall mission management, through navigation, to actual flight control, with outer loops depending on the correct functioning of the inner loops. Each of these loops requires different degrees of

[1] Note that this model is consistent with the dominant military command and control paradigm of Observe, Orient, Decide, and Act. (Boyd, 1987)

interaction with the operator. This necessitates a high degree of operator trust in the automation in the inner loops, because a single operator can only monitor the piloting of multiple vehicles, not control them manually. Previous studies show that if reliability decreased with more vehicles present, trust declined.   However, trust improved when the human operator was involved in planning and executing decisions. Even perfectly reliable automation, though, could not prevent a decrease in performance when workload increased (Cummings, Bruni, Mercier, & Mitchell, 2007).

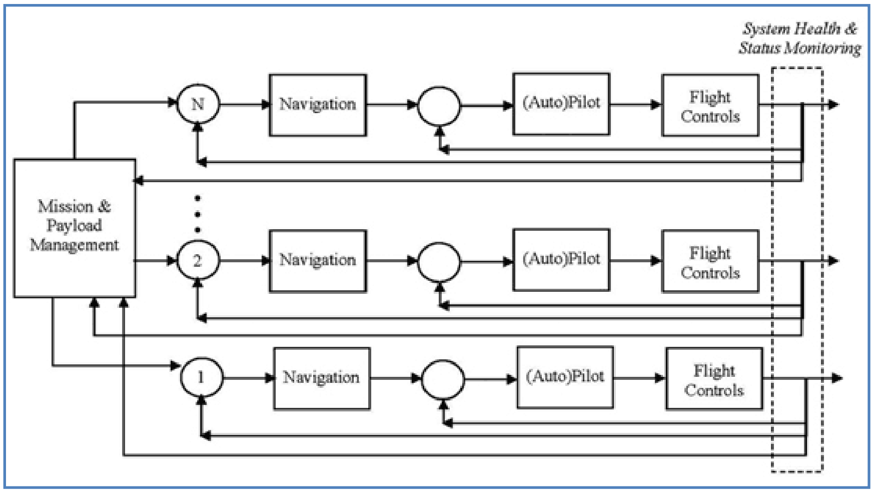


Figure 6 - Hierarchical Control Loops (Cummings, Bruni, Mercier, & Mitchell, 2007)

2.4.2 Display System Confidence

Context-aware systems may never attain 100% reliability since the context information that is needed to drive these systems is often incomplete, inaccessible, and/or uncertain. Therefore, the notion of trust between human operators and machines is necessary as operators tend to rely on certain properties of automation and develop a set of expectations of performance and behavior that can be greatly affected by the lack of a completely reliable system (Antifakos, Kern, Schiele, & Schwaninger, 2005). Antifakos, *et al.* propose displaying system confidence in order to help the human predict the outcome of the system and to use the system more effectively.

Antifakos *et al.* conducted a study to evaluate the effects of displaying system confidence on the user's trust towards a context-aware mobile phone. The study was conducted such that participants were evaluated on how much they relied on the context-aware systems. Different videos that varied in situation, criticality, and cue-availability, were presented to the participants who were asked 'In this situation, would you check the modality automatically selected by the system' and instructed to rate the question on a continuous scale from "no" to "yes". In the study, trust was measured by capturing how often the user would verify a modality the system automatically derived (Antifakos, Kern, Schiele, & Schwaninger, 2005).

The results of the study demonstrated that system confidence had a statistical impact on how the user relied upon the system. Figure 7 displays the results of the study. The graph portrays that participants verified the setting made by the context-aware system less often when the confidence of the system was medium or high. The verification rates tended to be higher with low system confidence, indicating that people inherently assume a confidence of above 0.5 and thus felt the need to verify the system more (Antifakos, Kern, Schiele, & Schwaninger, 2005).

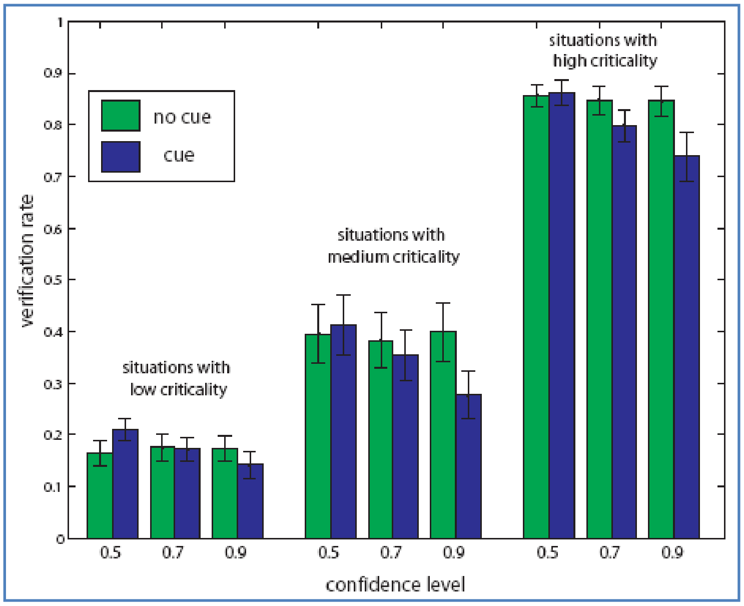


Figure 7 - Antifakos, Kern, Schiele & Schwaninger, 2005

2.5 Train for Appropriate Trust

Training, another HSI domain, is vital in conditioning operators to appropriately rely upon automation. As we have seen, operator trust in automation is largely based on expectations about the automation's ability, integrity, and benevolence. It is critically important, especially when establishing initial trust (which depends primarily on predictability), that expectations be defined realistically. This can be done through direct elicitation using various methods, such as vignette framing. In vignette framing, simplified scenarios are presented to expose participants to a particular situation or problem, and to elicit previously unstated expectations about the automation to be used in the scenario. These expectations can then be addressed, confirmed or adjusted as need be (Miller, 2008).

After initial expectations addressed, an operator must begin to interact with the system to develop proficiency. During this early phase, training from more experienced users can influence trust in the automation through social and organizational context, and direct experience will allow the operator to form an expectation of its reliability in various situations. This trust will be context-specific because user expectations are reinforced or refuted in specific situations. The operator's tendency to rely on the automation, or to act on a trust decision, will be based on uncertainty, time stress, and stakes.

Uncertainty refers to resolution, which in turn depends on the completeness of the user's understanding of system performance under given conditions. If training improves system knowledge, it will reinforce appropriate trust and reduce the amount of time an operator spends verifying the system (Cohen, Parasuraman, & Freeman, 1998).

Time stress, as the name implies, is the pressure to act quickly. When time stress is high, an operator will act quickly even when trust is uncertain. However, as discussed previously, under high time stress a user may choose not to rely on automation even if it is trusted, if that automation is too cumbersome to configure or engage. Training can improve this tendency, if the operator becomes familiar enough with the automation to reduce configuration time (Cohen, Parasuraman, & Freeman, 1998).

Stakes are consequences of a mistake. These mistakes can be caused by either through disuse by incorrectly rejecting the automation, or misuse by accepting a false recommendation. As training improves an operator's expectations of the system, trust should become more appropriate and the tendency of these errors should be reduced (Cohen, Parasuraman, & Freeman, 1998).

Training should account for the three dynamics of trust described in Section 2.2.3 as *Predictability*, *Dependability*, and *Faith*. In doing so, training should represent realistic system automation and operational scenarios. While designers and trainers cannot anticipate all possible situations an operator may face, experience with the system as close to its intended environment as possible will develop realistic expectations and engender appropriate trust in automation.

Chapter 3: Methodology

The literature review summarized in Chapter 2 provides a common frame of reference to model trust and its relationship to reliance and mission effectiveness. However, the multidisciplinary nature of trust and related concepts means that any trust model can be interpreted and critiqued by various communities in many different ways. While several models of trust have been proposed, these efforts have not produced a model that systems engineers can directly apply to integrate trust into system architectures such as those depicted within the DoD Architecture Framework, or DoDAF (DoDAF Version 1.5, 2007).

The intent of this research is to develop a model that can inform and guide systems engineering efforts through a deeper understanding of operator trust in automation, trust's influences, and trust's effect on reliance decisions. A second objective is to apply that model to an architecture for a specific purpose, namely UAV multi-mission management, to illustrate the model's utility in enhancing system design to improve mission effectiveness.

The resulting model is not intended to be understood from the perspective of an individual operator. Rather, it is a depiction of part of the system's state at a given time, where the operator is an integral part of the man/machine system. Thus, trust and other factors are properties of the system as a whole. Using this model, a systems engineering team should be able to make design decisions specifically to develop and maintain appropriate operator trust in the automation. They will be able to consider trust explicitly when developing information exchange and data requirements, and establish measures of performance and/or effectiveness to evaluate the appropriateness of trust.

Practical application of the model requires an adaptation of techniques developed by Majors Jonathan Zall and David O'Malley (2008) to incorporate cognition into the DoDAF. Together, these concepts may improve future single-operator multi-UAV system designs by appropriately calibrating operator trust in automation. Increased trust will improve mission effectiveness by reducing misuse and disuse of automation, allowing a more efficient distribution of effort between human and machine.

3.1 Macro View of Trust Relationships

The diagram shown in Figure 8 is useful for graphically summarizing the trust components and relationships. Context and operator perception combine to form expectation, the primary inputs to trust. Trust, as discussed previously, is an attitude that influences the reliance decision. However, factors external to trust may also influence reliance on automation in certain situations: ease of use, operating procedures (which might also be considered part of context), and likely others. A systems engineering model cannot include all possible influences on reliance, and henceforth the model being developed will not depict influences unrelated to trust. However, the readers should understand that trust, while an important factor, is not the sole determinant of a reliance decision.

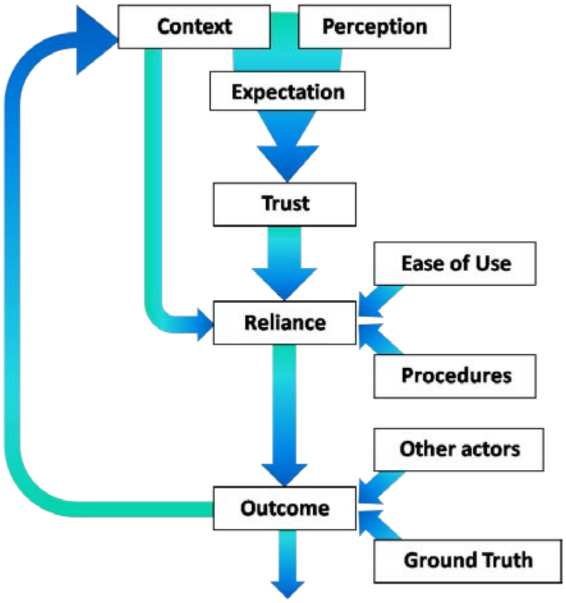


Figure 8 - Macro view of trust relationships

Reliance also influences the outcome of a particular decision. Yet, as reliance may depend on factors other than trust, the outcome of an event also can depend on factors external to the reliance decision. Other human, machine, and perhaps natural actors (e.g. birds) in the battlespace, differences between perception and ground truth, and other factors may also affect the outcome. These factors will not be depicted in the trust model intended for application in the systems engineering process. However, external factors can lead to a difference between apparent predictability and inherent predictability, which can incorrectly affect trust as discussed in section 2.2.3.1. Therefore, it is important for engineers and operators to evaluate whether anomalous outcomes result from an incorrect reliance decision based on inappropriate trust, from external factors, or from both.

It is particularly important during after-action analysis to determine when the operator accepted automation inappropriately (misused) or rejected automation inappropriately (disused). This information must be captured to the operator because achievement or failure of an expected outcome becomes part of an operator's context for future situations involving trust. When trust is appropriate and automation is relied on correctly, achievement of an effective and suitable outcome will reinforce that trust attitude. On the other hand, ineffectiveness may cause the operator to reassess the level of trust in the automation.

3.2 Development of the SE Model of Trust

While the macro view is useful for understanding the relationships between expectation, trust, reliance, and outcome, it has insufficient detail to be used when building systems architectures. To provide the requisite detail to allow systems and software engineers to consider trust explicitly within system architectures, the authors constructed a model using Unified Modeling Language (UML) notation. This structure is appropriate for Systems Modeling

Language (SysML) architectures also. The model will be described throughout the remainder of this chapter.

The UML model enabled the depiction of trust-related elements as object classes with associated attributes and methods. Attributes, depicted under the class name, are variables for each object of that class possesses. Methods, depicted in the bottom partition of each class, can be considered as functions performed by an object of that class. By depicting elements of the trust relationship as UML classes, these human cognitive activities may be represented as "trust objects" in architecture products. The UML classes may eventually provide linkages between trust objects and hardware or software objects representing components of system automation.

3.2.1 Perception

Chapter 2 developed a framework of trust and its dependencies. One of the major premises was that beliefs and perceptions are part of the information base that determines the trust attitude and affects operator reliance (Lee & See, 2004). The major concepts discussed in the literature were: situational awareness, self-confidence, time stress, and mental workload. For purposes of the current research, these concepts were classified as operator perceptions. Figure 9 was constructed to portray this relationship.

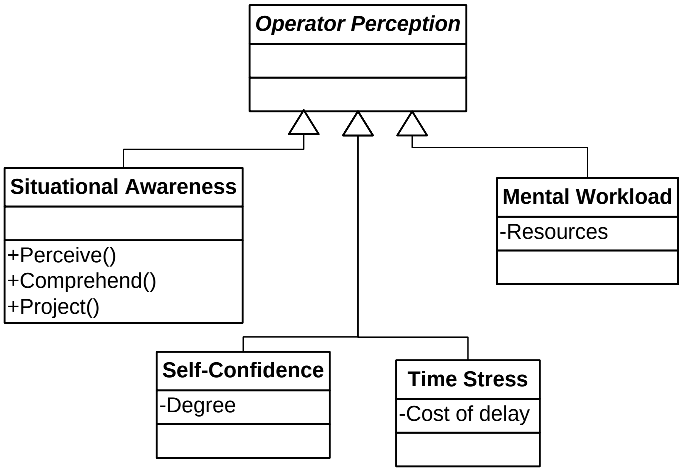


Figure 9 - UML Class Hierarchy of Operator Perception

Each of these perception constructs can be assessed empirically by designers to create automation that complements operator capabilities. However, an operator will behave based on their own mental model of each construct rather than objective measures of their own ability. By applying the model presented throughout this chapter to architecture, the authors believe designers will be able to shape operator perception to be more realistic to the automation's actual capabilities, as well as the operator's own abilities.

3.2.2 Context

Lee and See (2004) developed a model depicting the interaction of varying contexts as they relate to the evolution of trust: social, organizational, and individual. Bhattacharya, *et al.* (1998) also describes the environment and its inherent uncertainty as a context for the trust relationship. Those studies provided a basis for the UML diagram in Figure 10.

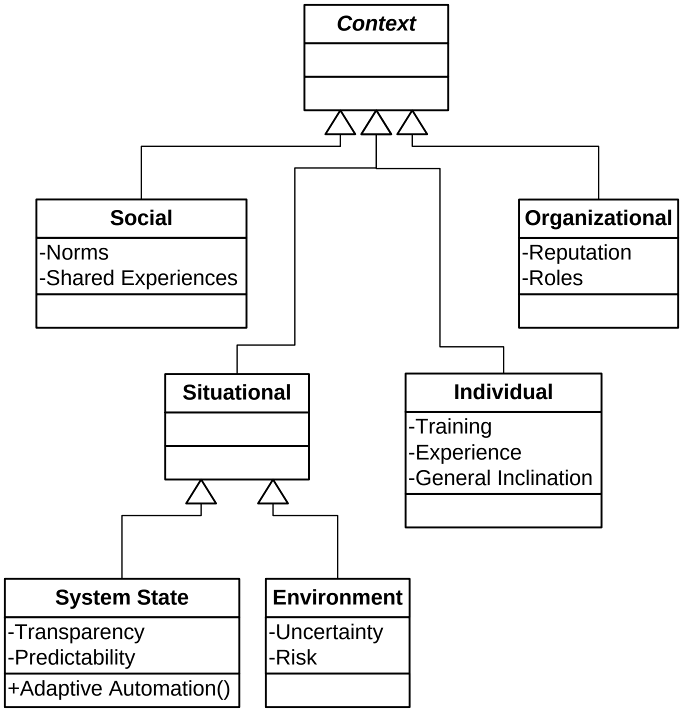


Figure 10 - UML Class Hierarchy of Context

In the class *Individual*, the following attributes were added: *Training, Experience,* and *General Inclination.* General inclination was defined as the inherent propensity to trust that varies across individuals. Therefore, this notion is an attribute specific to the subclass *Individual.* Likewise, training and experience with the system and automation are individual-specific properties, and therefore attributes of this subclass.

This research effort classified *Environment* and *System State* as subclasses of *Situational Context.* As outlined in Chapter 2, uncertainty is one of the main drivers for the need for trust where situational context plays a major role. Environments (considering both operational and local) constantly change during missions and the uncertainty and riskiness of the environment at any given point in time will ultimately impact an operator's trust in the automation.

Likewise, the current state of the system is situation dependent and must be consistent with the current needs of the operator. Otherwise, the system will not perform as expected, thus affecting the trust relationship. Transparency is defined as the amount of knowledge an operator has about why the automation performs as it does. The relationship between transparency and trust is not linear and therefore system planners must implement appropriate training to provide operators a degree of transparency, which can improve the accuracy of their expectations of system performance. However, appropriate transparency can also be designed into the system itself. For example, Chapter 2 describes how adaptive automation adjusts to the needs of the operator in a given situation, enhancing mission effectiveness by making the best use of human/machine collaboration.

3.2.3 Interaction of perception and context

This research differentiated the concepts of context and operator perception based upon their objective/subjective nature. Context refers to the objective components of the system and its environment while operator perception encompasses the subjective measures. Operator perception and context are interconnected through the expectations that an operator forms based upon the system context and the perceptions they develop in relation to the context and system automation. Figure 11 was developed to demonstrate this relationship.

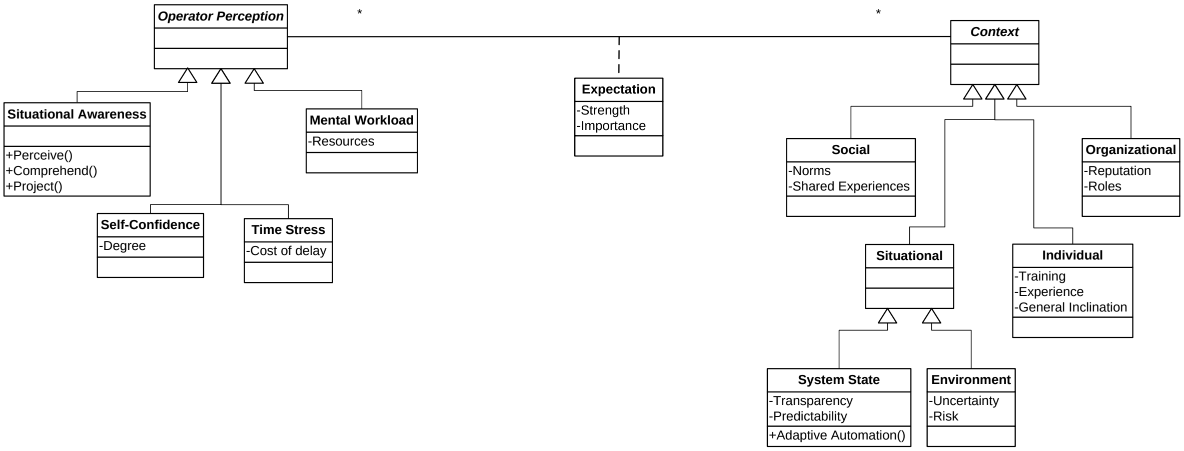


Figure 11 - Interaction of Perception, Context, and Trust

The various contexts and operator perceptions coalesce to develop a set of expectations that influence the trust relationship. For example, different contexts and levels of self-confidence will develop operator expectations on whether or not they feel they can operate better with or without system automation. In addition, situational context and operator perceptions will form a set of expectations on the level of operator input and control that is needed. For example, an operator flying a time-sensitive-target mission must deal with a rapidly changing environment in which his expectations may be biased toward anticipating events and acting decisively. This will likely cause him to operate the system differently than if he was conducting routine surveillance of stationary targets, in which the operator may expect the situation to be relatively stable, and any change that occurs need only be noted and not immediately acted upon. This example portrays the attribute of *importance* in the expectations that are developed, as where the prior example highlights the utility of the *strength* attribute of expectations. It also demonstrates the interplay of context and perception, in this case situational context and time stress.

3.2.4 Trust

Operator perceptions and context interact to form expectations. Expectations, in turn, form the primary inputs to trust. The UML diagram in Figure 12 demonstrates these interactions.

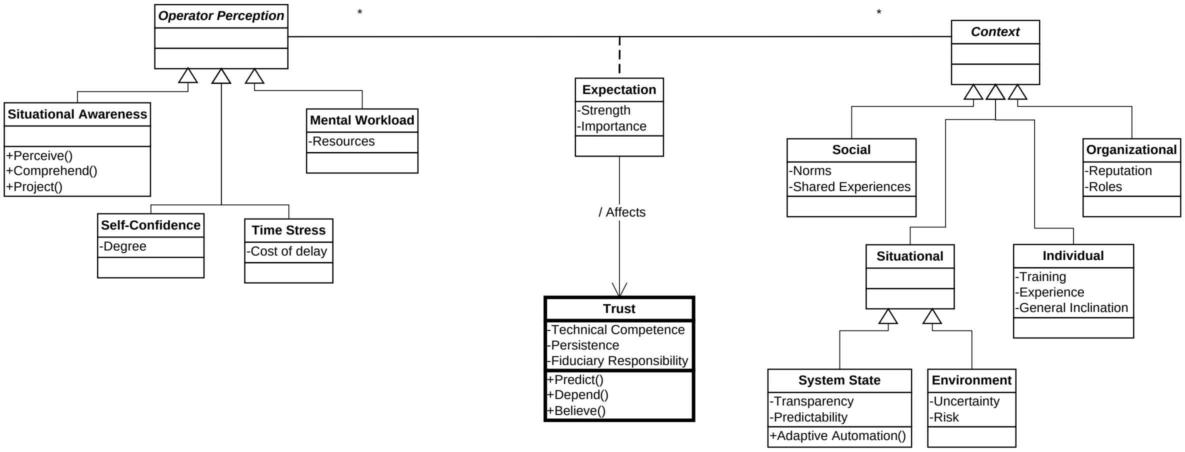


Figure 12 - Interaction of trust, perception, context and expectations

These relationships are consistent with Bhattacharya, *et al.*'s (1998) research, which determined that trust is formed in part by expectations, as well as Lee and See's work that portrayed context and perceptions as elements to evolution of trust. In addition, Lee and See characterized trust as an attitude to differentiate trust from the inputs of belief and information that form trust (Bhattacharya, Devinney, & Pillutla, 1998); (Lee & See, 2004).

Muir's trust bases, *Technical Competence, Persistence,* and *Fiduciary Responsibility*, provided the foundation of attributes for the class as they are the inherent characteristics that define the multidimensional nature of trust. In addition, Lee and See's (2004) dynamics of trust, *Predictability*, *Dependability*, and *Faith (*identified here as *Believe)*, act as the three fundamental methods, or stages, that an operator can have in his/her trust relationship (Muir, 1994).

3.2.5 Reliance Decision and its Outcome

An operator's trust in a system will be one of the main inputs into the decision of whether or not to rely on the system. The literature review in Chapter 2 defined three factors that result from the trust-reliance relationship. These were added as the main attributes of reliance: *Calibration, Resolution,* and *Specificity*. As described in Chapter 2, the goal for system planners is to design a system that fosters good calibration, high resolution, and high specificity. Varying levels of these attributes will lead an operator either to *Use*, *Misuse,* or *Disuse* the system. These are the three methods of reliance, where *Use* implies appropriate reliance, *Misuse* implies overreliance, and *Disuse* implies underreliance. These factors thus represent the outcomes of the reliance decision, as depicted in Figure 13.

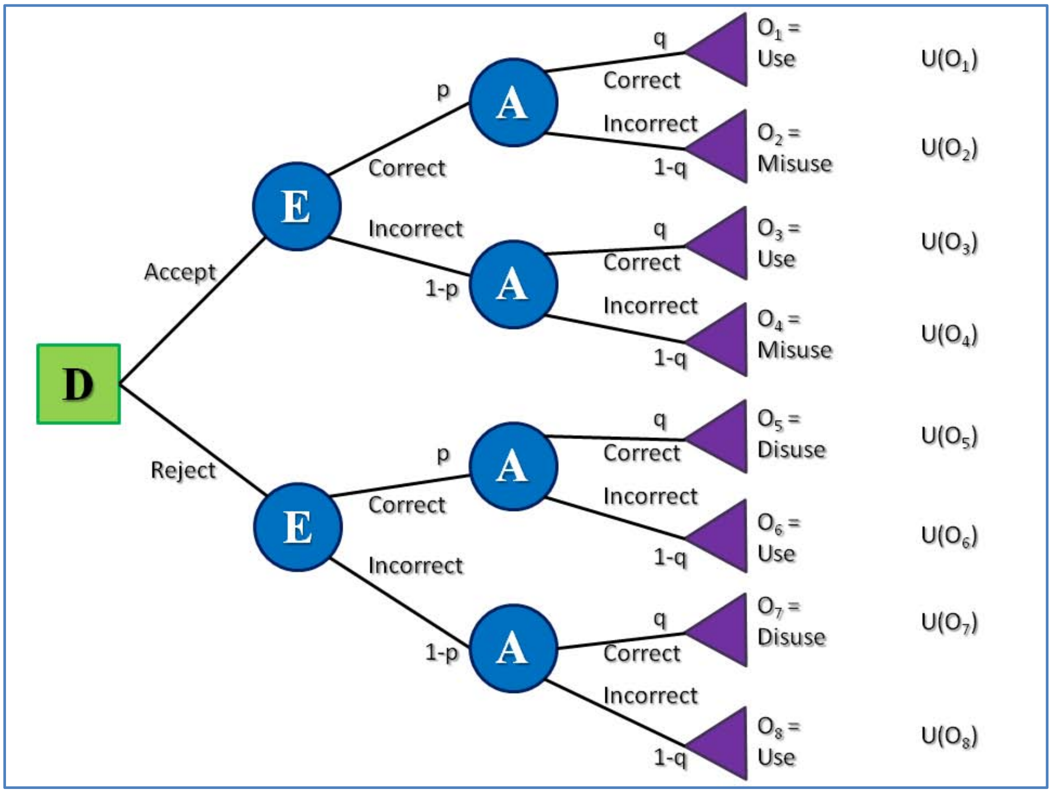


Figure 13 - Reliance Decision Tree

In Figure 13, D represents the *reliance* decision (i.e. accept or reject automation). Circles labeled E represent the operator's *expectation* of the automation's capability. As previously discussed, *expectation* has some degree of *strength*, shown as a probability p. Circles labeled A represent the actual capability of the automation, with some probability q that it is actually correct. In this model, it could be argued that the difference between expected capability p and actual capability q relates to the *calibration* of trust. Given the expected and actual capability, the possible outcomes of the reliance decision are represented as triangles. In forming expectations about possible outcomes, the operator will have mentally assigned each foreseeable outcome a value, although that value is often subjective. Shown here as $U(O_i)$ (where *i* represents the set of all foreseeable outcomes,) this value represents the utility, or *importance*, to the operator of achieving outcome O_i.

The *utility* of the actual outcome achieved, compared to the *utility* of the expected outcome, will shape the operator's future expectations and will ultimately affect the trust attitude. In other words, the degree to which the actual outcome matches the operator's expectation will affect the individual, situational, and/or organizational context for future iterations of this model. This change of context will reinforce or alter expectations and trust.

3.2.6 Integrated Model of Trust

In summary, this research effort combined all of the above concepts related to trust and reliance to develop the integrated model shown in Figure 14.

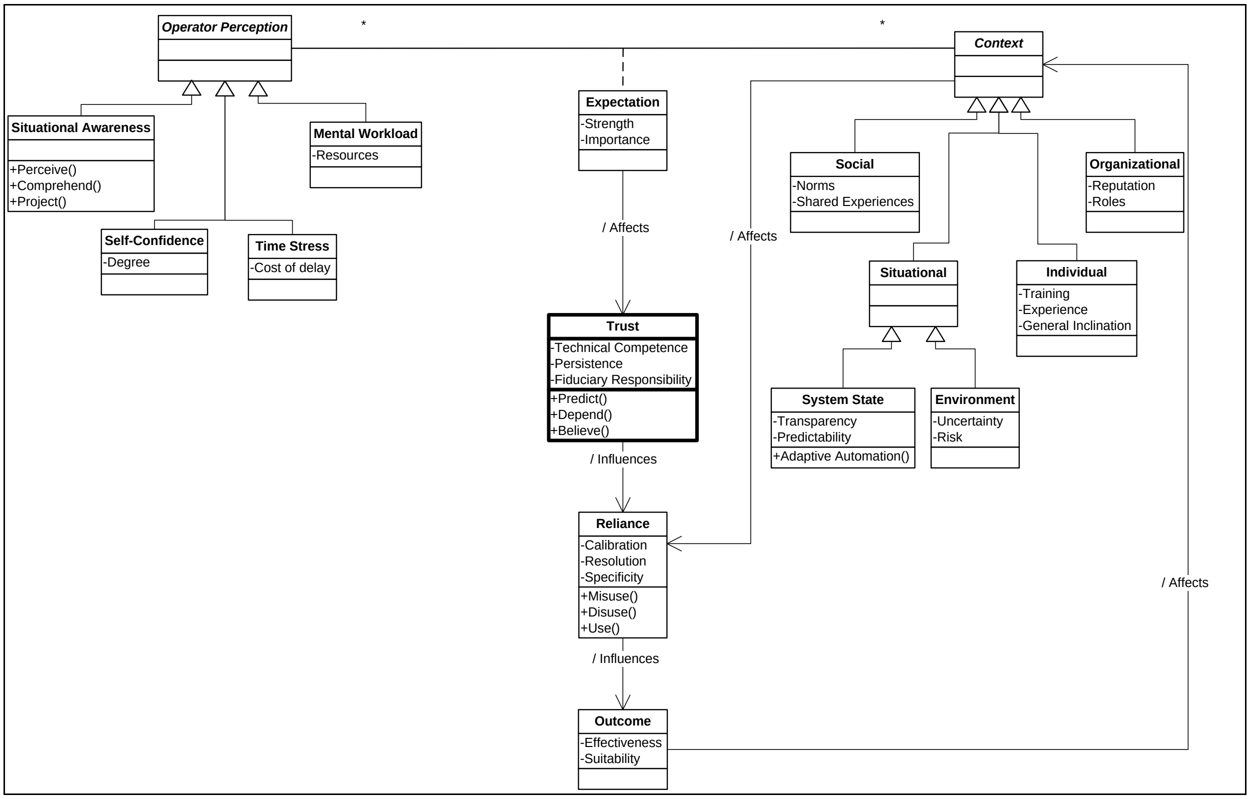


Figure 14 - SE Model of Trust Relationships

The main intent of this research is to provide a means of architecting trust early in system design in order to develop systems that will increase mission effectiveness. Therefore, effectiveness and suitability are the main attributes of the class *Outcome.* These measures develop the foundation for a user to evaluate the appropriateness of their trust in the system. As a whole, this model serves as a guide to explicitly address trust within system architectures. Architectures can be developed in many ways, but those of relevance to military planners generally follow the DoD Architecture Framework, or DoDAF.

3.3 DoD Architecture Framework

3.3.1 Application of Trust Model to System Architecture

The trust model described in this chapter is derived from a large body of work by researchers in human factors, psychology, human systems integration, and other related disciplines. The model is intended to aid systems engineers to improve the effectiveness of system design by considering trust relationships and their related tasks, information requirements, and processes explicitly during architecture development. Therefore, the model is in line with the HSI community's emphasis to integrate human capabilities into system design beginning with conceptualization and continuing through system disposal. In order to do so, the trust model must be applicable to existing architecture frameworks, both at the conceptual level and as elements of specific products. This integration will be discussed in the next chapter.

3.3.2 DoDAF v1.5

The DoD Architecture Framework "provides the guidance and rules for developing, representing, and understanding architectures based on a common denominator across DoD, Joint, and multinational boundaries." DoDAF version 1.5 consists of four types of views: Operational View (OV), Systems and Services View (SV), Technical Standards View (TV), and All View. Applied correctly, these views form an integrated architecture, which means that the different views are not independent representations of separate parts of the system. Rather, "architecture data elements are uniquely identified and consistently used across all products and views within the architecture" (DoDAF Version 1.5, 2007). The individual views, then, are different ways to represent aspects of the system in order for members of interdisciplinary subsystem engineering teams to integrate their components with the overall effort.

The OV mainly represents the different tasks and activities that the system performs along with operational nodes and the associated information exchanges. In accomplishing this representation, the views are designed to convey the types of information exchanged, the frequency of exchange, the tasks and activities supported by the information exchange, and the nature of the information exchanges. The SV is designed to portray the system, service, and interconnection functionality providing for, or supporting operational activities. Therefore, many aspects of the systems views are linked back to artifacts with the operational views. The TVs are comprised of a set of rules that provide the necessary framework to establish engineering guidelines and product lines to ensure the system is built to satisfy the specified set of operational requirements. The AV is designed to establish the scope and context of the architecture. The AV is comprised of the overarching aspects that relate to all three views including products such as vision statements, CONOPS, scenarios, etc. (DoDAF Version 1.5, 2007). The major relationships among the types of views are illustrated in Figure 15 below.

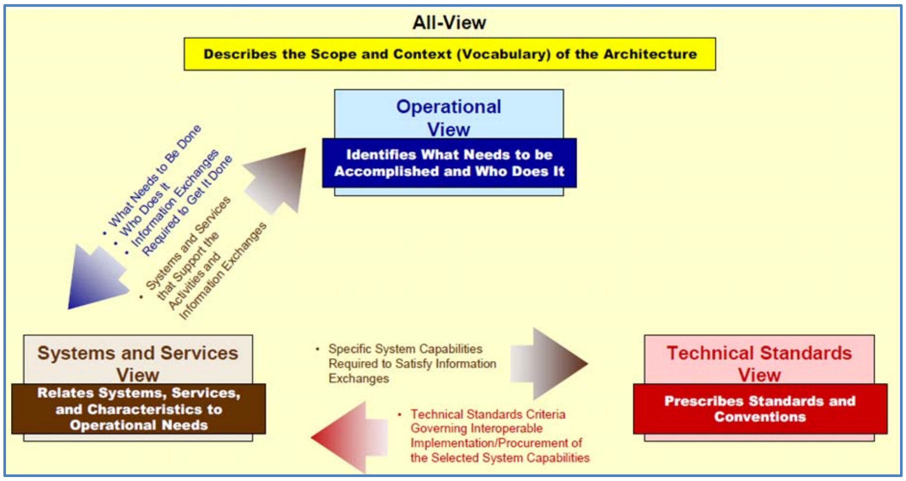

All-View
Describes the Scope and Context (Vocabulary) of the Architecture
Operational View
Identifies What Needs to be Accomplished and Who Does It
• What Needs to Be Done
• Who Does It
• Information Exchanges Required to Get It Done
• Systems and Services that Support the Activities and Information Exchanges
Systems and Services View
Relates Systems, Services, and Characteristics to Operational Needs
• Specific System Capabilities Required to Satisfy Information Exchanges
• Technical Standards Criteria Governing Interoperable Implementation/Procurement of the Selected System Capabilities
Technical Standards View
Prescribes Standards and Conventions

Figure 15 - Fundamental Linkages Among the Views, DoDAF v1.5 Vol. 2

In all, there are 28 distinct views that define the system, its components, and their interactions. The applicable views, their product name, and a brief description of each are outlined in Table 1.

Table 1 - List of Views in DoDAF v1.5

Applicable View	Frame-work Product	Framework Product Name	General Description
All View	AV-1	Overview and Summary Information	Scope, purpose, intended users, environment depicted, analytical findings
All View	AV-2	Integrated Dictionary	Architecture data repository with definitions of all terms used in products
Operational	OV-1	High-Level Operational Concept Graphic	High-level graphical/textual description of operational concept
Operational	OV-2	Operational Node Connectivity	Operational nodes, connectivity, and information exchange need lines between nodes
Operational	OV-3	Operational Information Exchange Matrix	Information exchanged between nodes and the relevant attributes of that exchange
Operational	OV-4	Organizational Relationships Chart	Organizational, role, or other relationships among organizations
Operational	OV-5	Operational Activity Model	Capabilities, operational activities, relationships among activities, inputs, and outputs; overlays can show cost, performing nodes, or other pertinent information
Operational	OV-6a	Operational Rules Model	One of three products used to describe operational activity—identifies business rules that constrain operation
Operational	OV-6b	Operational State Transition Description	One of three products used to describe operational activity—identifies business process responses to events
Operational	OV-6c	Operational Event-Trace Description	One of three products used to describe operational activity—traces actions in a scenario or sequence of events
Operational	OV-7	Logical Data Model	Documentation of the system data requirements and structural business process rules of the Operational View
Systems and Services	SV-1	Systems and Services Interface Description	Identification of systems nodes, systems, system items, services, and service items and their interconnections, within and between nodes
Systems and Services	SV-2	Systems and Services Communications Description	Systems nodes, systems, system items, services, and service items and their related communications laydowns
Systems and Services	SV-3	Systems-Systems/ Services-Systems/ Services-Services Matrix	Relationships among systems and services in a given architecture; can be designed to show relationships of interest, e.g., system-type interfaces, planned vs. existing interfaces, etc.
Systems and Services	SV-4a	Systems Functionality Description	Functions performed by systems and the system data flows among system functions
Systems and Services	SV-4b	Services Functionality Description	Functions performed by services and the service data flow among service functions
Systems and Services	SV-5a	Operational Activity to Systems Function Traceability Matrix	Mapping of system functions back to operational activities

Applicable View	Frame-work Product	Framework Product Name	General Description
Systems and Services	SV-5b	Operational Activity to Systems Traceability Matrix	Mapping of systems back to capabilities or operational activities
Systems and Services	SV-5c	Operational Activity to Services Traceability Matrix	Mapping of services back to operational activities
Systems and Services	SV-6	Systems and Services Data Exchange Matrix	Provides details of system or service data elements being exchanged between systems or services and the attributes of that exchange
Systems and Services	SV-7	Systems and Services Performance Parameters Matrix	Performance characteristics of Systems and Services View elements for the appropriate time frame(s)
Systems and Services	SV-8	Systems and Services Evolution Description	Planned incremental steps toward migrating a suite of systems or services to a more efficient suite, or toward evolving a current system to a future implementation
Systems and Services	SV-9	Systems and Services Technology Forecast	Emerging technologies and software/hardware products that are expected to be available in a given set of time frames and that will affect future development of the architecture
Systems and Services	SV-10a	Systems and Services Rules Model	One of three products used to describe system and service functionality—identifies constraints that are imposed on systems/services functionality due to some aspect of systems design or implementation
Systems and Services	SV-10b	Systems and Services State Transition Description	One of three products used to describe system and service functionality—identifies responses of a system/service to events
Systems and Services	SV-10c	Systems and Services Event-Trace Description	One of three products used to describe system or service functionality—identifies system/service-specific refinements of critical sequences of events described in the Operational View
Technical Standards	TV-1	Technical Standards Profile	Listing of standards that apply to Systems and Services View elements in a given architecture
Technical Standards	TV-2	Technical Standards Forecast	Description of emerging standards and potential impact on current Systems and Services View elements, within a set of time frames

3.3.3 DoDAF v2.0

The draft DoDAF version 2.0 (DoDAF Version 2.0, 2008) adds additional types of views, but emphasizes the architecture process rather than the generation of specific products. V2.0 describes its focus as follows: "The central core of DoDAF v2.0 is a data-centric approach where the creation of architectures to support decision-making is secondary to the collection, storage, and maintenance of data needed for efficient and effective decisions. The architect and stakeholders select views to ensure that architectures will explain current and future states of the process or activity under review" (DoDAF Version 2.0, 2008). Accordingly, further discussion of system architecture in this research will concentrate on process rather than specific products, although products may be used to illustrate important points. For reference, Figure 16 demonstrates the evolution of DoDAF views in version 2.0 and provides a mapping of DoDAF v1.5 views to the new framework.

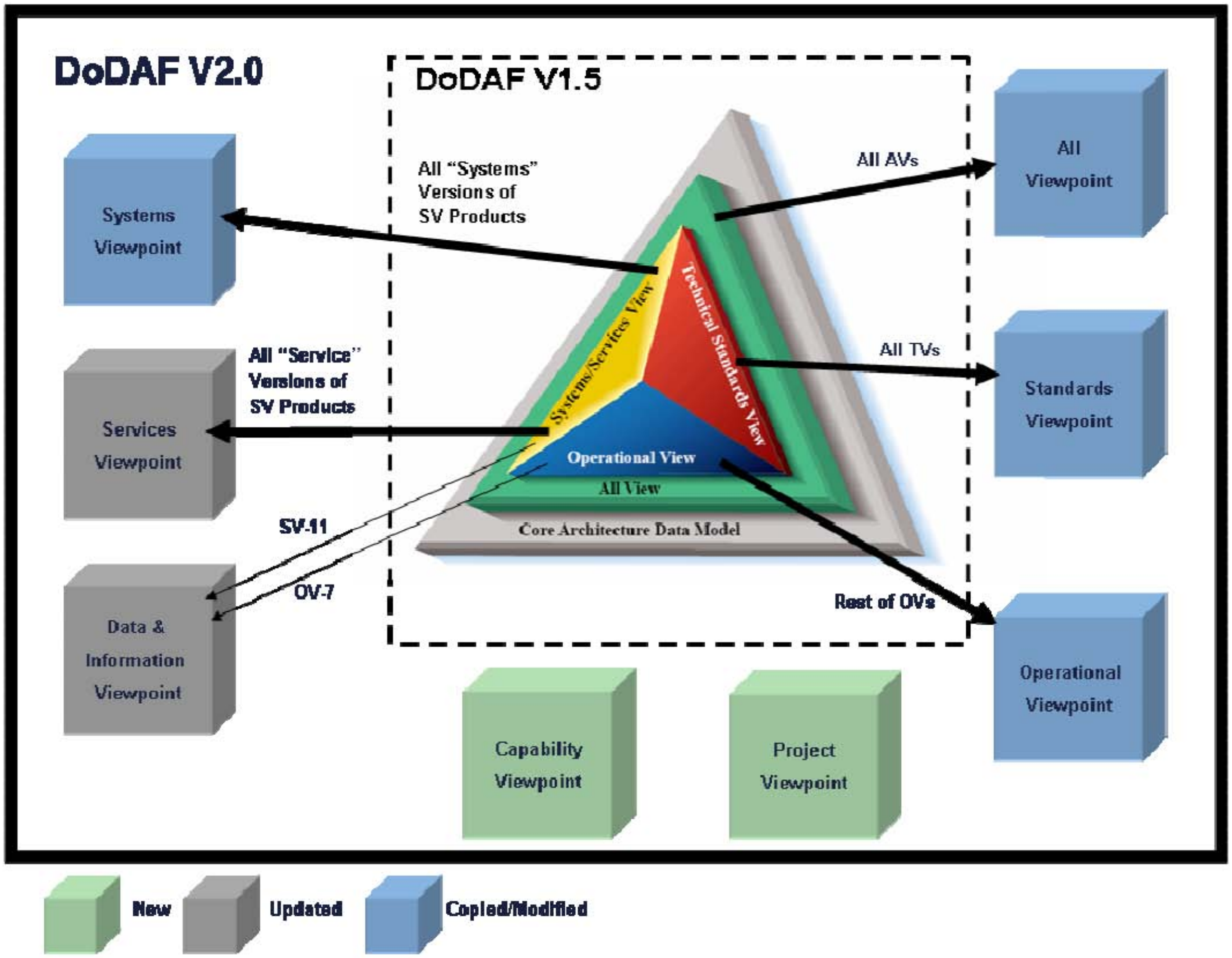


Figure 16 - Notional Mapping of DoDAF v1.5 Views to Draft DoDAF 2.0 Views

3.4 Zall/O'Malley Model

3.4.1 Architecting Cognition

Research conducted by Zall and O'Malley (2008) focused on integrating human cognition into DoD systems architecture. They identified five pertinent areas of cognition that provided the foundation for early system trades, and manpower, personnel, and training (MPT) decisions. These five areas are: Cognitive Tasks, Cognitive Inputs, Cognitive Outputs, Cognitive Roles, and Cognitive Environments (Zall & O'Malley, 2008).

One of the main drivers behind Zall and O'Malley's research was the development of cognitive tasks (an identification of which type of cognition is required) and allocation of the task to either human or machine; i.e. either cognitive or pseudo-cognitive (CPC) tasks. These allocations in early architecture provide the framework for system developers and planners to better understand and incorporate varying levels of automation and human supervisory control into systems.

Cognitive inputs are the specific information that the cognitive tasks require to be completed while the cognitive output is the consequential set of information that results from the cognitive task (Zall & O'Malley, 2008). These two types of information can depict the necessary enabling items and system complexity when outlined in the system's architecture to aid the developer/planner in identifying and avoiding potential HSI shortfalls.

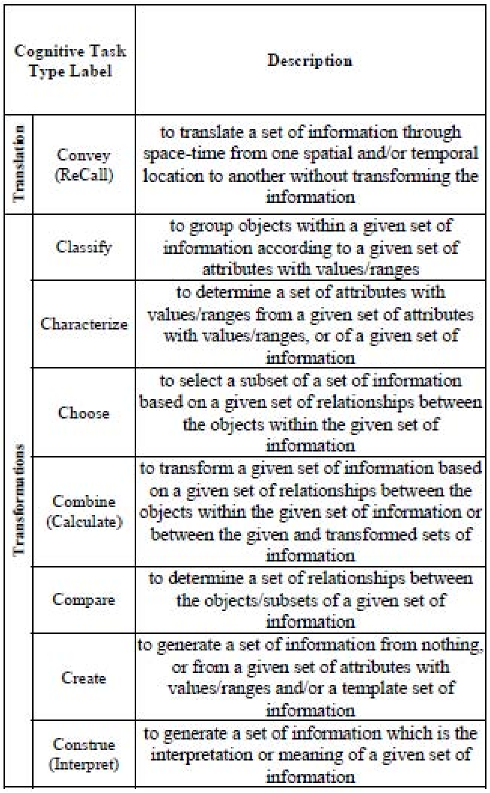
The cognitive role evolves from a specified set of cognitive tasks and is intended to take on a specific personnel position description to help define MPT system requirements (Zall & O'Malley, 2008). The premise of architecting this aspect is to ensure that personnel are utilized to their maximum potential and are not under/over utilized by allocating too few/many cognitive tasks to the specific cognitive role an operator will fill.

Lastly, architecting the cognitive environment provides a necessary representation of the interactions different roles and entities will have on one another. In addition, the cognitive environment presents an opportunity to account for any design related constraints that MPT requirements may place on the system such as whether or not an operator has enough space to properly work.

3.4.2 Fundamental CPC Tasks

Zall and O’Malley used an atomic approach to decompose the many cognitive task types into a smaller set of fundamental tasks or “bases” that can be combined to form any type of cognitive activity. These bases are outlined in Table 2.

Table 2 - Early Conceptual Design Eight Fundamental CPC Tasks (Zall & O'Malley, 2008)



Cognitive Task Type Label		Description
Translation	Convey (ReCall)	to translate a set of information through space-time from one spatial and/or temporal location to another without transforming the information
Transformations	Classify	to group objects within a given set of information according to a given set of attributes with values/ranges
	Characterize	to determine a set of attributes with values/ranges from a given set of attributes with values/ranges, or of a given set of information
	Choose	to select a subset of a set of information based on a given set of relationships between the objects within the given set of information
	Combine (Calculate)	to transform a given set of information based on a given set of relationships between the objects within the given set of information or between the given and transformed sets of information
	Compare	to determine a set of relationships between the objects/subsets of a given set of information
	Create	to generate a set of information from nothing, or from a given set of attributes with values/ranges and/or a template set of information
	Construe (Interpret)	to generate a set of information which is the interpretation or meaning of a given set of information

The focus of their research was to integrate these elements of cognition into existing DoDAF products. Because the cognitive tasks primarily represent activities, Zall and O’Malley

(2008) focused their efforts on expanding OV-5 activity diagrams to include all activities down to the cognitive level. This process was developed to help identify the cognitive tasks in a system and assist engineers in allocating these tasks to either human or machine.

3.5 Architecting for Appropriate Trust

As previously described, the trust relationship includes activities and data elements. Thus, while the techniques developed by Zall & O'Malley (2008) provide a starting point from which to architect trust, they are insufficient to encapsulate the full trust relationship due to their focus on activities. Therefore, in addition to developing objects within the architecture views, the various trust dependencies will be evaluated as data elements and information exchanges identified during the architecture process.

To determine methods for architecting trust, it is necessary to evaluate a host of views from a particular framework. In accomplishing this task, DoDAF version 1.5 architecture products will be used; however, it is necessary to reiterate that the application of a *process*, not the generation of *products*, is the enduring goal of this research. The primary focus will be to develop a strategy to aid systems engineers and planners in considering the trust relationship in system design. This emphasis on process, rather than products, will be consistent with DoDAF version 2.0.

The final portion of our research will evaluate the architecture products associated with a particular developmental UAS. The goal of this evaluation is to demonstrate a strategy that can be applied to any type of system within any architecture framework which will provide maximum applicability for systems engineers.

Chapter 4: Process for Integrating Trust Within DoDAF

4.1 The DoDAF Architecture Process

The integration of trust into system architecture should not be an afterthought or corrective action, but rather should occur throughout the architecture development process. Accordingly, the basic processes for integrating trust will coincide with those of the larger architecture effort. DoDAF v2.0 describes a six-step architecture development process, as follows (DoDAF Version 2.0, 2008).

Step 1: Determine Intended Use of Architecture. In this step, the architect defines the purpose and intended use of the architecture, and establishes the architecture strategy for the project. This is generally based on information provided by the process owner.

Step 2: Determine Scope of Architecture. Scope refers to the depth and breadth of the architecture. In this step, the architecture's problem set and context are defined, and the architect begins to define the level of detail required for the architecture content. It is important for the architect to define a clear and appropriate scope that enables an expected result but avoids excessive breadth. For example, often only the manmade system components are addressed.

Step 3: Determine Data Required to Support Architecture Development. The scope defined in step 2 influences the level of detail needed for each data entity and attribute. Here, data refers to both the information needed for execution of the process, and other data needed to produce the desired process changes (e.g. documentation about the architecture effort itself.) The data content is recorded as attributes, associations, and concepts that will later be mapped to specific architecture views as needed.

Step 4: Collect, Organize, Correlate, and Store Architecture Data. In this step, architects collect and organize data in order to develop a structure for the overall architecture effort. The data are organized into a taxonomy that allows identification of data gaps, as well as process and services required.

Step 5: Conduct analyses in support of architecture objectives. This step prepares the architecture for approval by the process owner by validating the architecture effort against established process owner requirements. If changes are required from the validation process, they are made by iterating steps 3 through 5 as necessary.

Step 6: Document Results in Accordance with Architecture Framework. The final step involves the creation of actual architecture views as needed, based on the underlying data developed in the previous steps. DoDAF v2.0 provides a number of predefined “DoDAF-described Views”, including many of those contained in DoDAF v1.5 and similar frameworks. The DoDAF does not require specific views to be used, but encourages architects to use DoDAF-defined views where applicable. However, DoDAF v2.0 also provides user-defined views, or 'Fit-for-purpose Views', which can be created in non-standard formats as long as they are consistent with the DoDAF meta-model. This allows architects to tailor products to their specific project or organization, or to support their key decision-maker.

4.2 Architecture Methodologies

The execution of the six-step architecture development process requires the use of analytical techniques to organize and portray system elements. These generally fall into two classes of approaches: Structured Analysis and Object-Oriented Analysis.

Structured Analysis is a process-oriented approach that centers on hierarchical decomposition of processes, along with corresponding inputs, outputs, controls, and mechanisms (ICOMs). Structured Analyses may result in process data flow diagrams, which allow users to visualize how specific data inputs trace to given actions; process task-dependency diagrams, which depict the flow of material, information, or a service through all its steps in a logical or required order; and entity-relation models, which describe the structure of system domain data types and the rules that govern system data (DoDAF Version 2.0, 2008).

Object-Oriented Analysis (OOA), describes the operational use cases, places data in the context in which it will be used, and provides traceability for system and software design. OOA consists of activity diagrams, which depict dynamic system behavior; class and object diagrams, which portray sets of objects (i.e. performers) and their relationships as classes (object types). Another useful diagram in Object Oriented Analyses is the sequence diagram, showing the temporal changing nature of interacting objects (DoDAF Version 2.0, 2008).

These approaches are not mutually exclusive, and integrated architectures often will make use of both techniques, manifested in different views to demonstrate various aspects of the system. The trust model developed in Chapter 3 is applicable within both techniques, depending on the purposes for which it is needed in specific parts of the architecture process.

4.3 Architecting Trust

The emphasis on process, rather than products, in DoDAF v2.0 provides a standard framework within which to develop system architectures, while allowing system architects flexibility in creating the architecture itself. When considering trust within the framework, the six steps of the DoDAF process can be paired into three stages: Determine Scope and View; Determine and Organize the Data; and Analyze and Document Results. Portions of the trust model developed in chapter 3 can be applied during each of these stages, as will be described throughout the remainder of this section. In addition, the authors have prepared a summary of this section as a quick reference guide for system architects and technical managers. This summary can be found as Appendix A.

4.3.1 Determine Scope and View

The Joint Capabilities Integration Development System (JCIDS) (Joint Chiefs of Staff, 2007) process provides system planners with a high level view of the requirements needed to fulfill an existing gap in the Department of Defense. As a new system enters into the formal acquisition process, systems engineers and planners have a basic understanding of the requirements that are needed to fulfill the gap; however, few system-specific details exist at this stage in the process. This overarching JCIDS view of the required capability provides systems engineers and planners a basis to begin to consider the relationships that arise in system design and then begin to plan for incorporating trust into the human-system relationships.

Within the current architecture framework, the main views present at this stage would be the AV-1 Overview and Summary Information, and the OV-1 High-Level Operational Concept Graphic. These views provide planners with graphical and textual representations of how the system will be implemented. The architect can begin to incorporate trust using these views by determining how and where an operator may interact with the system. These interactions form the foundation of the trust relationship. By identifying the foundation of the trust relationship, the architect should be able to focus subsequent development efforts where they will most improve trust. In addition, the AV-1 provides insight into the nature of the expected operational environment of the system. The operational conditions are important in establishing the situational context portion of the trust model. Likewise, the uncertainty and risk inherent in the operational environment are key factors in establishing operator expectations, which will drive the trust relationship between the operator and the automation.

Also within this stage, planners should consider the manpower, personnel, and training (MPT) requirements needed to operate the system. By shaping MPT requirements early in the development process, planners can account for the influence of *organizational context* (e.g. command structure, roles) and *individual context* (e.g. training, familiarity with a given interface) of their intended user, rather than reacting to the influences post hoc.

Knowledge of context may affect design decisions, or a specific design implementation may inspire changes to training. In either case, early architecture steps are useful to begin to determine scenarios where operator trust in automation will be particularly important and to define corresponding design requirements to foster appropriate trust within those operating scenarios.

4.3.2 Determine and Organize the Data

System architects focus much effort on determining and collecting data elements necessary to define roles and activities within the system. Determining and collecting data elements allows planners to apply the trust lens, by determining which trust bases and dependencies are relevant within the system, under what scenarios they are critical, and how to effectively integrate and portray the elements.

During the determination and collection of data steps, architects should focus on providing *transparency* of the system's state to the operator in a manner that allows operators to form accurate expectations, and then reinforce appropriate trust. This might be accomplished by displaying confidence information about automated functions (Antifakos, Kern, Schiele, & Schwaninger, 2005); (Jamieson, Wang, & Neyedli, 2008) by adapting the behavior of automation in order to account for operator perceptions, like SA or mental workload (Cummings

& Mitchell, 2008); (Parasuraman, Barnes, & Cosenzo, 2007), or other implementation-specific means. As the system architecture is developed and refined, architects can account for trust while establishing specific operating scenarios and making design decisions.

There are four main categories of architecture views that systems engineers and architects will be concerned with: functional/behavioral, informational, sequence/temporal, and structural. Each of these categories allows system architects to portray different aspects of the trust relationships within their system.

4.3.2.1 Functional / Behavioral

Functionally decomposing a system will allow planners and designers to obtain a more in-depth understanding of the tasks the system will perform and the human and machine system nodes that are essential in accomplishing them. Information exchanges between system nodes directly impact certain trust dependencies and therefore must be accounted for during system design. For example, knowledge of the system state (e.g. system transparency) will facilitate more appropriate operator-automation trust if the underlying data is displayed effectively (Konstantinou, Liagkou, Spirakis, Stamatiou, & Yung, 2005).

In addition, an understanding of the inputs, controls, outputs, and mechanisms (ICOMs) that drive system activities will be pertinent in the creation and/or modification of data elements that will aid designers in architecting for appropriate trust. For example, one of the recommendations from earlier research detailed in Chapter 2 was to display system confidence to the operator (Antifakos, Kern, Schiele, & Schwaninger, 2005). This activity may require the addition of a new task with new information exchanges between system components.

Two of the main views in DoDAF v1.5 that focus on functional elements are the OV-5 Operational Activity Model, and the SV-4 Systems Functionality Description. The OV-5 portrays activities and the ICOMs that relate to them. As mentioned in Chapter 3, Zall and

O’Malley (2008) outlined a process to incorporate fundamental cognitive tasks in OV-5 diagrams. Their main focus was to enhance the views in relation to the MPT requirements by adding ICOMs and tasks, and classifying them as cognitive (human) or pseudo-cognitive (machine). The differentiation of cognitive and pseudo-cognitive tasks with ICOMs implies interaction between human operators and automation, and thus the presence of trust relationships that need to be considered during architecture development. In addition, as Zall and O'Malley discussed, the allocation of cognitive and pseudo-cognitive tasks provides a basis to develop tailored training that enhances relationships between actors in the system. This training could develop and/or reinforce the individual context component of the trust model, and could provide input to allow the operator to form expectations of the system's behavior under realistic operational conditions.

4.3.2.2 Informational

Portrayals of system information exchanges are increasingly important in the architectures of complex systems. Since trust is inherently dependent on the information the trustor has about the trustee, the environment, etc., the depiction of informational flows in the system will be particularly relevant to trust. Consideration of elements such as situational context may necessitate additional information flows, due to their effect on forming and reinforcing operator expectations, trust, and reliance.

Informational elements typically are depicted in views such as the SV-6 Data Exchange Matrix, which "specifies the characteristics of the system data exchanged between systems. This product focuses on automated information exchanges that are implemented in systems" (DoDAF Version 1.5, 2007). Likewise, the OV-7 Logical Data Model "describes the structure of an architecture domain’s system data types and the structural business process rules that govern the system data" (DoDAF Version 1.5, 2007).

4.3.2.3 Sequence / Temporal

The development of time-phased portrayals of the system activities provides the foundation for engineers to determine the desired dynamics and level of trust for anticipated situations. In essence, sequence diagrams will foster the development of the type and level of automation that is most effective for the different scenarios and environments that the system will be expected to operate in. As anticipated events and desired outcomes are outlined in given scenarios, the criticality and time stress of each can be assessed in order to design proper automation.

The information derived from sequence diagrams will provide the basis for designers to evaluate where and when to use adaptive automation as outlined in Chapter 2. This type of automation will be imperative in some highly critical and time stressed situations where operator *faith* in the automation is necessary to accomplish the objectives. In addition, planners can utilize this information to build their training scenarios in order to provide realistic operator experience, which will improve the operator's ability to *predict* system behavior and reinforce realistic expectations about the system's *dependability*.

The main DoDAF views that depict sequence information are the OV-6c Operational Event-Trace Description, and the SV-10c Systems Event-Trace Description. The OV-6c is a time-phased portrayal of the operational activities to accomplish a mission, while the SV-10c captures the changing system state over time including the data exchanges that occur between system components. Additional data exchanges such as the receipt and acknowledgement of a command directly following an operator command will likely result in an enhancement in the relationship between the operator and automation. As mentioned in Chapter 1, the workload of current Global Hawk operators is unnecessarily increased due to the need to check system responses to ensure the automation received the command. Additional information exchanges would alleviate this problem and help foster the transition to one operator multiple UAS systems.

4.3.2.4 Structural

An understanding of all the subsystems that comprise the overall system and their interfaces will provide the information necessary to visualize the interconnections and their relationships. One of the main goals of the Human Factors Engineering domain within the HSI community is to create effective integration of human-system interfaces to achieve optimal total system performance (INCOSE, August 2007). These elements portray automation, the human, and what relationships exist between them. With respect to trust, knowledge of system interdependence is paramount in determining whether or not additional interfaces are needed to facilitate the trust relationship. For example, to provide verification of commands and display system confidence, the systems that provide information must interface with each other. In addition, as illustrated in the trust model developed in chapter 3, the operator's knowledge of the system state, combined with perceptions such as SA, affect trust by influencing expectations about the system's predictability and dependability.

The architecture views that contain structural type of information are the SV-1 Systems Interface Description and the SV-3 Systems-Systems, Services-Systems, Services-Services Matrices. The SV-1 provides a view of all the systems and services and their interconnections, while the SV-3 details the relationships and interfaces between the systems and services. Any tasks, activities or system functions that were added to the OV-5, SV-4, OV-6c, SV-10c, etc. to reflect trust relationships must be accounted for in the SV-1 and SV-3.

4.3.3 Analyze and Document Results

The last two steps in the architecture development process seek to validate the output of the previous steps and to create architecture products based on the validated data. It is important to note that the data requirements definition, data gathering, analysis (validation), and documentation (product generation) steps are iterative processes. As the system architecture matures, relationships between system nodes may change, which may require reassessment of the activities and information exchanges required to support appropriate operator-automation trust.

4.4 Practical Application of the Trust Lens

A conceptual look at a trust-integrated architecture development process illustrates that it is possible to address trust explicitly within system architectures. By doing so, engineers may affect changes to system design to reinforce innate trust relationships, and theoretically, raise overall system effectiveness by shaping the operator's trust in automation in ways that improve the speed and accuracy of man/machine collaboration. Moreover, this can be done regardless of the actual architecture framework employed or specific products used to define it.

However, a theoretical examination of the architecture development process cannot demonstrate these conclusions definitively. To do so, it is necessary to examine the architecture of a real world system, apply the trust integration process described, and assess the results for indicators of improved system effectiveness. This research will now apply the theoretical study to the Battlefield Airman Tactical Camera (BATCAM) UAS.

4.4.1 Intro to BATCAM/CUSS Architecture

The Air Force Institute of Technology (AFIT) is currently conducting research in multiple UAS areas. Several current masters' students at AFIT are conducting research in CUSS to optimize control of multiple UAVs for varying tasks including, but not limited to: formation manipulation, optimization of coverage area, flight path deconfliction, and sensor placement optimization. The platform being used for this research is the BATCAM UAS. This ongoing research provides opportunity for the current authors to apply the trust relationships to an existing UAS infrastructure.

4.4.1.1 System Overview

The BATCAM UAS consists of a hand launched air vehicle that uses GPS and autopilot algorithms for navigation and landing, with commands (e.g. waypoints, altitude, etc.) uploaded from a ground control station using a PC-based interface. Sensor data is downloaded to the same control station. The air vehicle is 61cm in length, has a 53cm wingspan and weighs approximately 0.38kg. Its size and weight support man portable backpack operations. The BATCAM's operations are limited due to its size, with a ceiling of 300m, a range of 3km and an endurance of 18 minutes. The vehicle carries forward and side looking cameras in a removable pod to provide real-time situational awareness and targeting information to the operator. The system was developed by the Munitions Directorate of the Air Force Research Laboratories and has been in inventory since 2003. Figure 17 shows the BATCAM air vehicle. (Parsch, 2005)

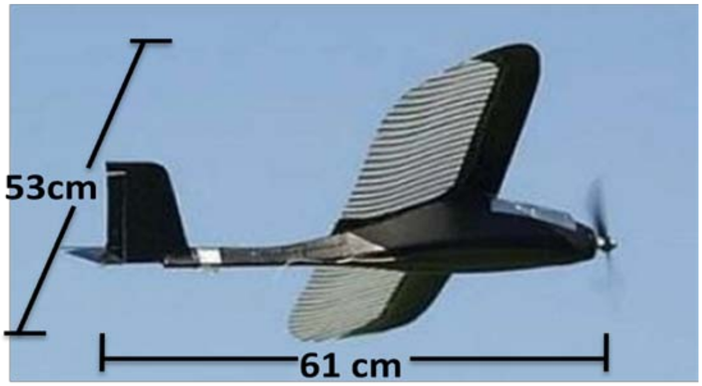


Figure 17 - BATCAM UAV (Parsch, 2005)

Cummings, *et al.* (2007) developed a set of hierarchical control loops to describe a set of basic mission functions UASs must perform (see Figure 6 in this document.) Currently, the BATCAM operators are involved in Mission & Payload Management, Navigation, and to some degree the Pilot functions. However, due to lack of reliable automation there is high demand upon the operator to maintain SA and verify that the air vehicle is performing as expected. To decrease operator workload and facilitate a transition to a "one operator, multiple UAS system," a high degree of trust must be established between the operator and system automation. This gap may present an opportunity for the current authors to apply the trust concepts outlined in the beginning of this chapter.

4.4.1.2 Autopilot

The BATCAM UAV uses the Kestrel autopilot system, provided by Procerus Technologies. Procerus offers the OnPoint software for use with the Kestrel. OnPoint is a target-tracking program capable of providing geo-referenced track data for stationary and moving target. (Procerus Technologies, 2006)

Figure 18 - Autopilot System (Procerus Technologies, 2006)

4.4.1.3 Ground Station

Two components which comprise the ground station: the Ground Control Unit (GCU) and the Portable Computing Device (PCD). The BATCAM uses a single GCU which integrates an RF modem, video receiver, and analog to digital video converter. Analysis of the GCU revealed little internal coupling between these three components, with integration providing only a single form factor. (Sakryd & Ericson, 2008)

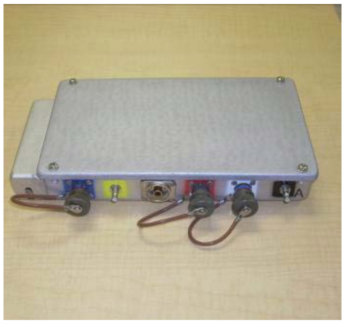


Figure 19 - Ground Control Unit (Sakryd & Ericson, 2008)

The PCD is the computing device which integrates most of the system's functionality. The BATCAM uses a Panasonic Tough Book™ which has been ruggedized for field use. (Sakryd & Ericson, 2008)

Figure 20 - Portable Computing Device (Sakryd & Ericson, 2008)

4.4.1.4 Operations and Employment Concept

The BATCAM UAS primary employment include the following scenarios survey a stationary target, survey a moving target, reconnoiter ahead of a moving vehicle, provide surveillance of a series of waypoints, conduct a broad area search, and conduct search for a target. Each of these scenarios envisions employment of up to four UAVs operating cooperatively to perform the assigned mission, controlled by a single operator. This research will limit its focus to the following BATCAM operations: surveying a stationary target and providing surveillance of a series of waypoints.

4.4.1.4.1 Surveil a Stationary Target BATCAM Operational Process

A user wishes to surveil a stationary target. The user deploys the system if it is not already in use. The user inputs the location of the target into GCU. The GCU directs one or more UAVs to the target location. The UAVs transmit sensor data to the user. Upon reaching the specified location, the UAVs execute a "loiter" flight pattern to provide sensor coverage on the target. The user designates the target on his sensor monitoring screen. The UAVs maintain sensor coverage of the target. At the end of the mission, the UAVs return to their previous task or to their launching base.

The stationary target scenario may also be entered from another surveillance task. The user monitoring a UAV sensor feed identifies a fleeting target of further interest. The user directs the system to monitor the target. The UAV transitions to a "loiter" flight pattern around the selected target.

The user may change the default loiter distance to try to get better sensor resolution on the target or to minimize the chance of detection of the UAV.

4.4.1.4.2 Provide Surveillance of a Series of Waypoints BATCAM Operational Process

A user wishes to provide surveillance of a series of waypoints, such as a road, route, perimeter, maritime transit lane, or geographic border, usually to provide sensor coverage ahead of a convoy. The user deploys the system if it not already in use. The user inputs the designated waypoints with the GCU. The GCU directs the UAVs to initial waypoints. The system organizes the coverage of the UAVs in accordance with the programmed waypoints. The surveillance continues until the UAVs are recalled or redirected by the user.

This scenario may be entered from another surveillance task. The user detects a route from a UAV sensor feed. The user uploads waypoint data with the GCU. The user commands the corresponding UAV to follow the route.

During the mission, if a user detects a point for further study, the user commands the corresponding UAV to focus on that point. The UAV transitions to a “loiter” flight plan around the selected target. The system redistributes the coverage ahead of the convoy among the remaining UAVs. When the user directs the system to stop monitoring the target, the system directs the UAV that was designated for the selected target back into formation ahead of the convoy and redistributes coverage.

4.4.2 Architecting Trust: Determining BATCAM Purpose and Scope

The current thesis group working with the BATCAM focused their efforts on categorizing and defining the system. The following diagram is an OV-1 that was developed to illustrate the operational capability of the BATMCAM system.

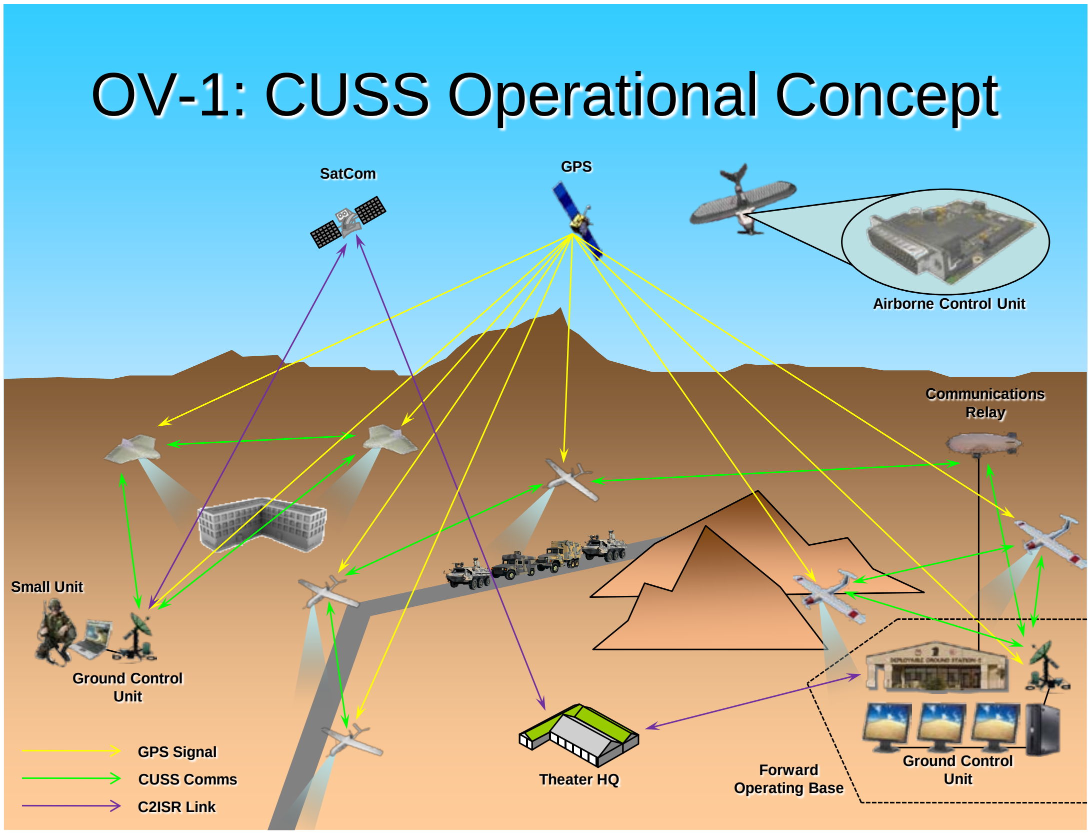


Figure 21: BATCAM Operational Concept

Section 4.3.1 outlined how early system architecture products can help facilitate initial recognition of where and how an operator may interact with system automation as well as the different interfaces that potentially exist throughout the system. From this diagram, a system architect can conclude that the interaction between an operator and the automation will occur at the Ground Control Units (GCU), and that the system will interface with the environment at the airframe and the GCU. The automation will be facilitated by the CUSS software and the communication links between the GCUs, communications relay, and airframes. This illustration also indicates additional interfaces such as: GPS – GCU, GPS – airframe, SatCom – GCU, and SatCom - Theater Headquarters. The architect can use this data to direct early application of trust dependencies to improve system performance through appropriate trust.

4.4.3 Trust Data Required to Support BATCAM Architecture Development

4.4.3.1 Functional/Behavioral

The functional/behavioral architecture views can also foster more appropriate reliance if trust relationships are considered. The AFIT BATCAM research group has developed OV-5 and SV-4 views that support this category. As outlined in Section 3.5.2.2, one of the main objectives for the BATMCAM UAS is to provide surveillance of a series of waypoints. Figure 22 portrays the main operational tasks that are required to fulfill this objective.

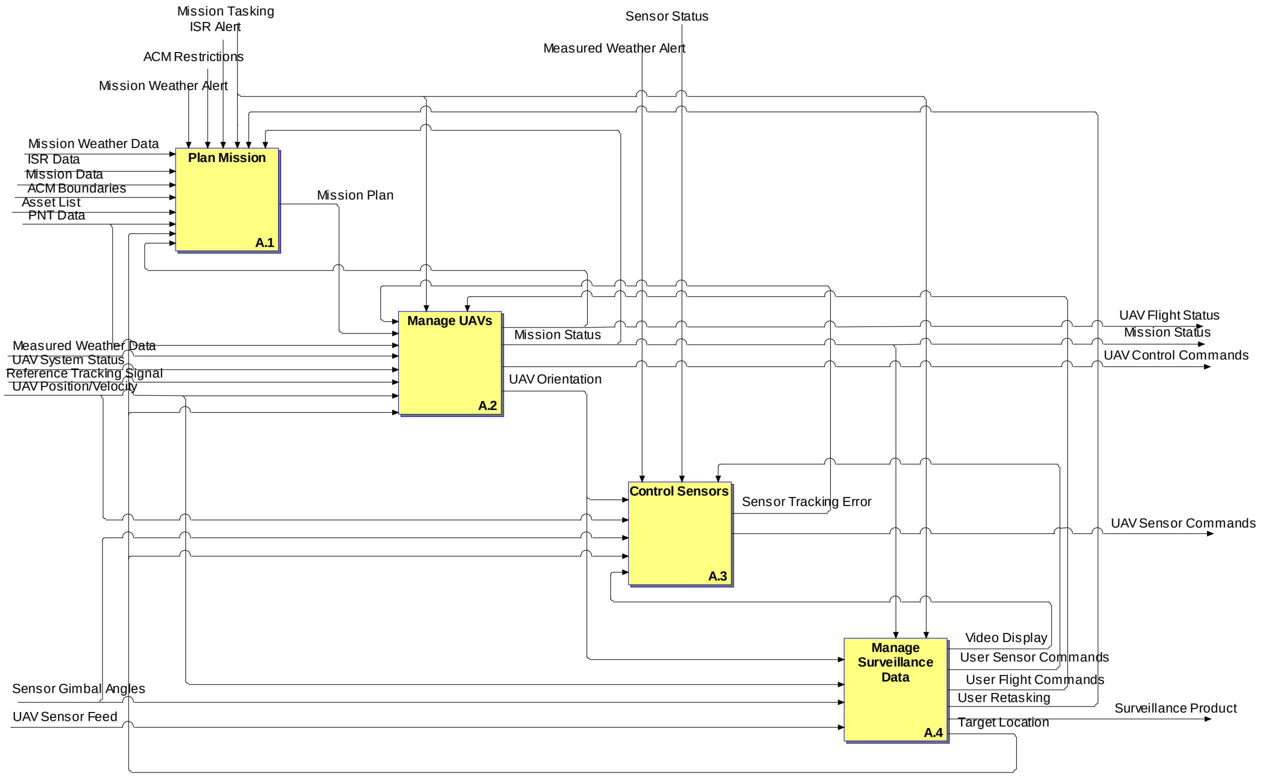


Figure 22 - Provide Surveillance OV-5 Diagram

This is the initial OV-5 operational diagram that was developed to describe the *Provide Surveillance* operational activity. It is necessary to add additional ICOMs to Figure 21 in order to design the system for appropriate trust. *Displaying system confidence* and *acknowledging operator commands* are two recommendations that were made in Section 4.3.2.1 that are applicable to the BATCAM UAS and expected to induce more appropriate reliance. These two tasks would be represented as outputs from the *Manage UAVs* task. Figure 23 shows the OV-5 updated to reflect these two trust factors.

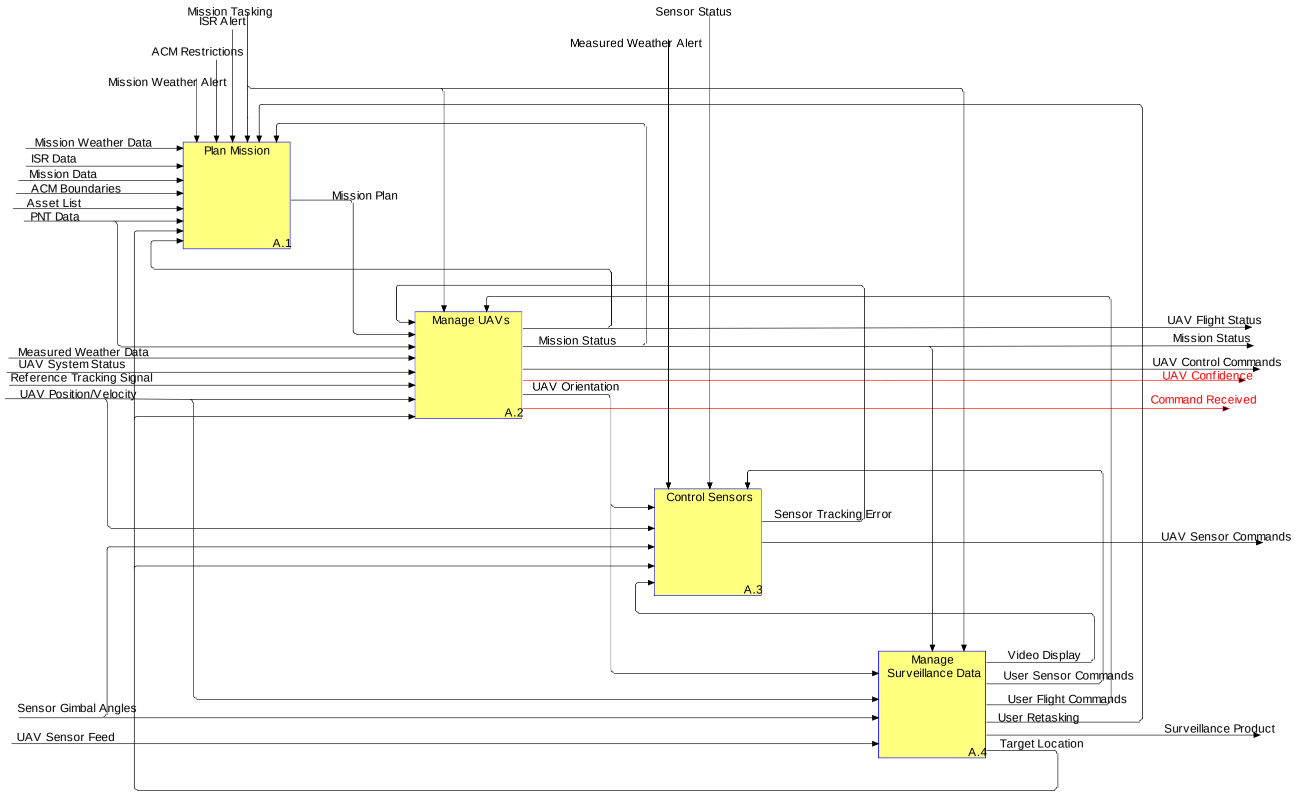


Figure 23: OV-5 Diagram with Trust Factors Added

In order to obtain a more in-depth understanding of where these tasks are derived from and how they affect system design and architecture, it is necessary to functionally decompose the system to the appropriate level. As the diagram above portrays, the outputs originate from the task *Manage UAVs.* Figure 24 portrays the functionally decomposed task before the two trust factors were included.

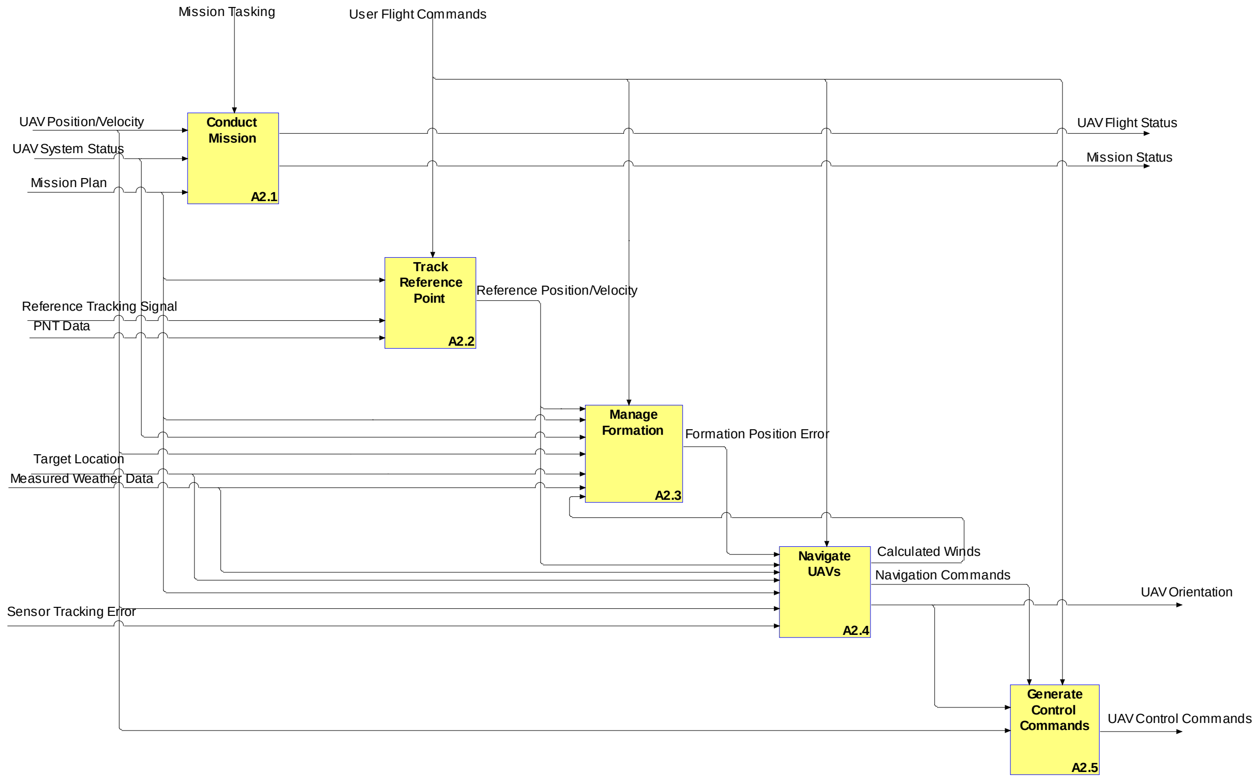


Figure 24 - Manage UAVs OV-5 Activity Diagram

The two trust factors, *Acknowledge Commands* and *Determine Confidence*, were added as new operational tasks under the *Manage UAVs* functional decomposition. The main input for acknowledging commands is *User Flight Commands*. Therefore, this ICOM was added as an input to the *Acknowledge Commands* task. The result of this task is the output *Command Received.* In order to determine UAV confidence, many ICOMs are needed as inputs. The authors of this research added the following ICOMs as input to the new task: *UAV Position/Velocity*, *UAV System Status*, *Reference Tracking Signal*, *Formation Position Error*, *Calculate Winds*, and *UAV Orientation.* Figure 25 portrays the OV-5 diagram with the addition of the trust factors.

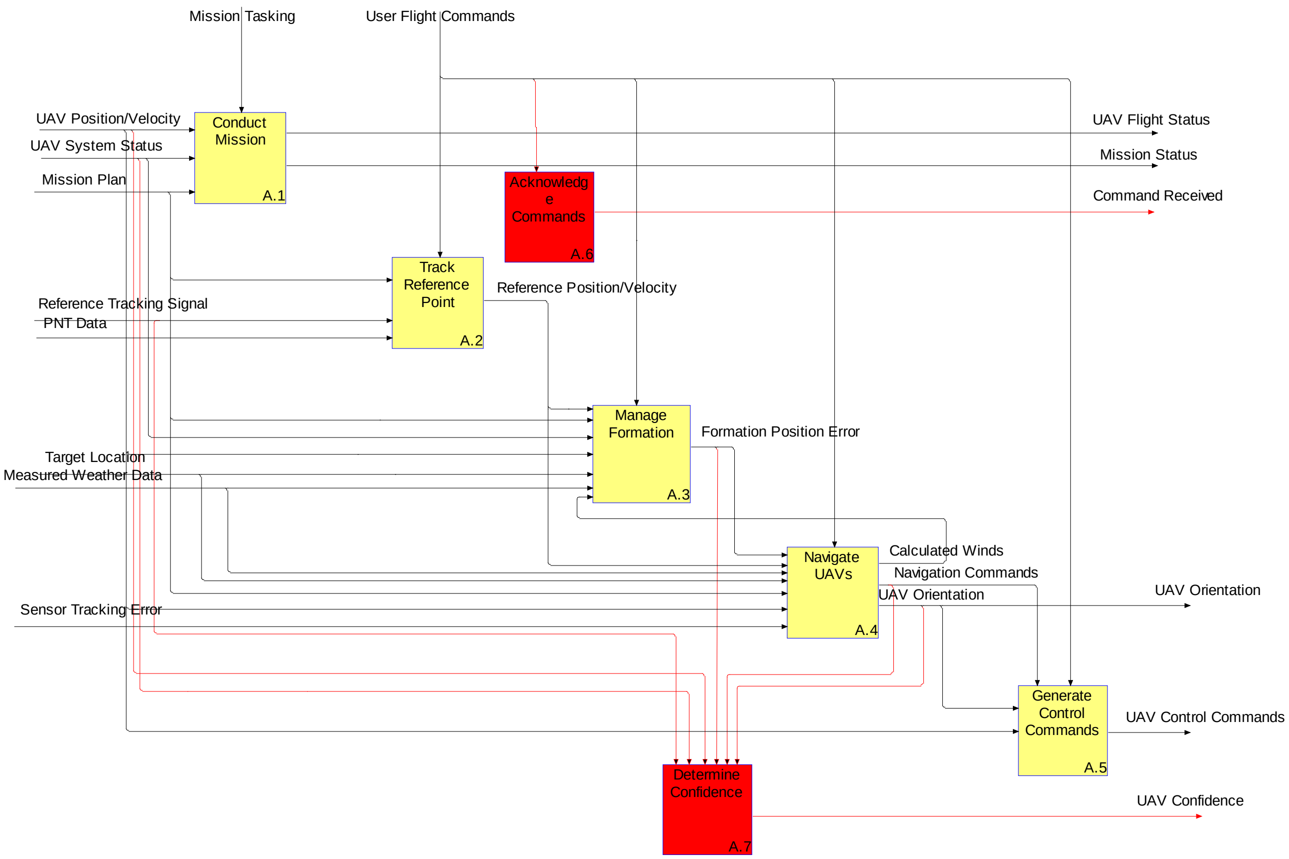


Figure 25: Updated OV-5 Diagram with Trust Factors

The SV-4 diagrams depict the functions that are performed by systems captured in the OV-5 diagrams. For an activity to be accomplished, a system must be charged with performing the task or a part of the task. Therefore, the SV-4 Systems Functionality Description diagrams must be updated to reflect changes in the OV-5 Operational Activity diagrams. Figure 26 represents the architecture before the trust factors were included.

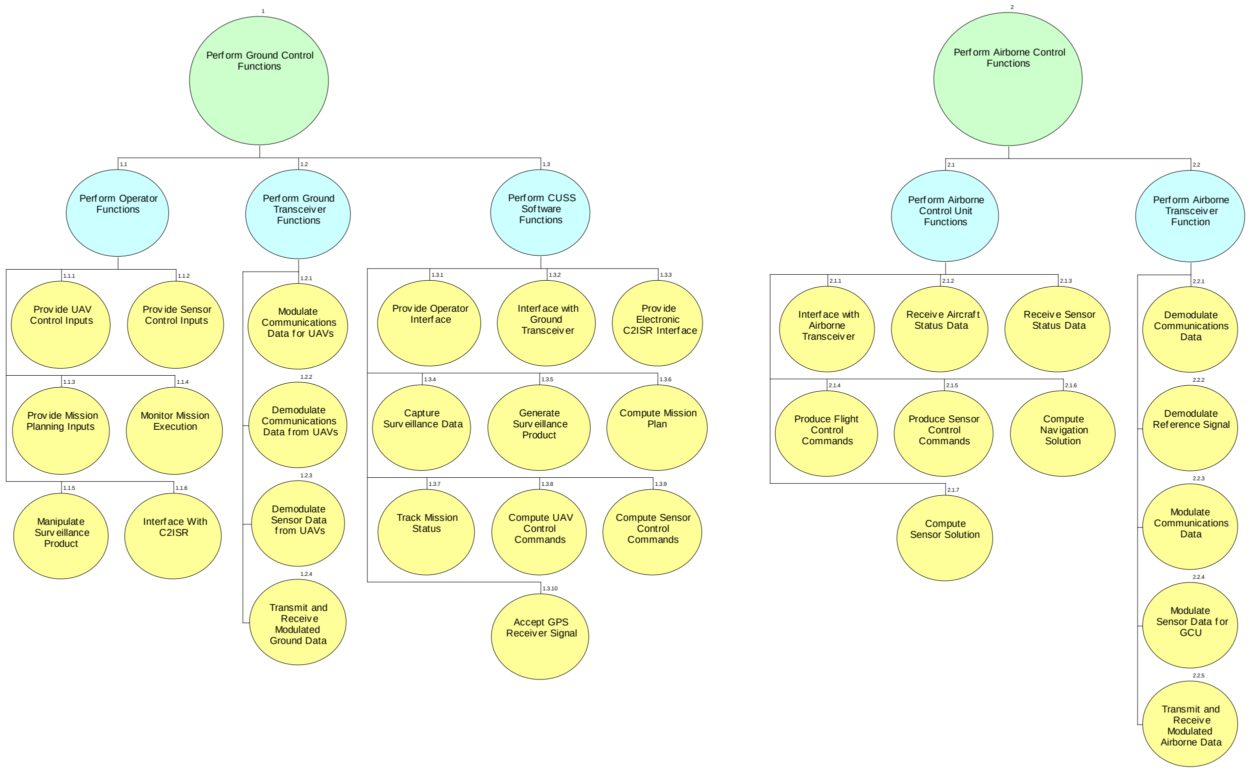

1
Perform Ground Control Functions
1.1
Perform Operator Functions
1.2
Perform Ground Transceiver Functions
1.3
Perform CUSS Software Functions
1.1.1
Provide UAV Control Inputs
1.1.2
Provide Sensor Control Inputs
1.1.3
Provide Mission Planning Inputs
1.1.4
Monitor Mission Execution
1.1.5
Manipulate Surveillance Product
1.1.6
Interface With C2ISR
1.2.1
Modulate Communications Data for UAVs
1.2.2
Demodulate Communications Data from UAVs
1.2.3
Demodulate Sensor Data from UAVs
1.2.4
Transmit and Receive Modulated Ground Data
1.3.1
Provide Operator Interface
1.3.2
Interface with Ground Transceiver
1.3.3
Provide Electronic C2ISR Interface
1.3.4
Capture Surveillance Data
1.3.5
Generate Surveillance Product
1.3.6
Compute Mission Plan
1.3.7
Track Mission Status
1.3.8
Compute UAV Control Commands
1.3.9
Compute Sensor Control Commands
1.3.10
Accept GPS Receiver Signal
2
Perform Airborne Control Functions
2.1
Perform Airborne Control Unit Functions
2.2
Perform Airborne Transceiver Function
2.1.1
Interface with Airborne Transceiver
2.1.2
Receive Aircraft Status Data
2.1.3
Receive Sensor Status Data
2.1.4
Produce Flight Control Commands
2.1.5
Produce Sensor Control Commands
2.1.6
Compute Navigation Solution
2.1.7
Compute Sensor Solution
2.2.1
Demodulate Communications Data
2.2.2
Demodulate Reference Signal
2.2.3
Modulate Communications Data
2.2.4
Modulate Sensor Data for GCU
2.2.5
Transmit and Receive Modulated Airborne Data

Figure 26 - BATCAM SV-4 Functional Decomposition Diagram

The two main activities that were added need to be allocated to functions are "displaying system confidence" and "acknowledging operator commands". The CUSS software is the system that is best suited to compute the system confidence, and hence will be added as a function underneath it. The Computing Device will display the system confidence to the user. This system is currently not shown and needs to be added as a sub-function under *Perform Ground Control Functions.* It is important to take into account the system display because past research efforts have shown that the manner that system confidence is displayed has a direct impact on operator reliance upon the automation. (Jamieson, Wang, & Neyedli, 2008) Determining the best method for displaying data is left as a recommendation for further study in Chapter 5. Finally, acknowledging operator commands is a function that must originate from the aircraft control unit and therefore is added under *Perform Airborne Control Unit Functions.* Figure 27 and Figure 28 represent the BATCAM SV-4 diagrams with trust factors added.

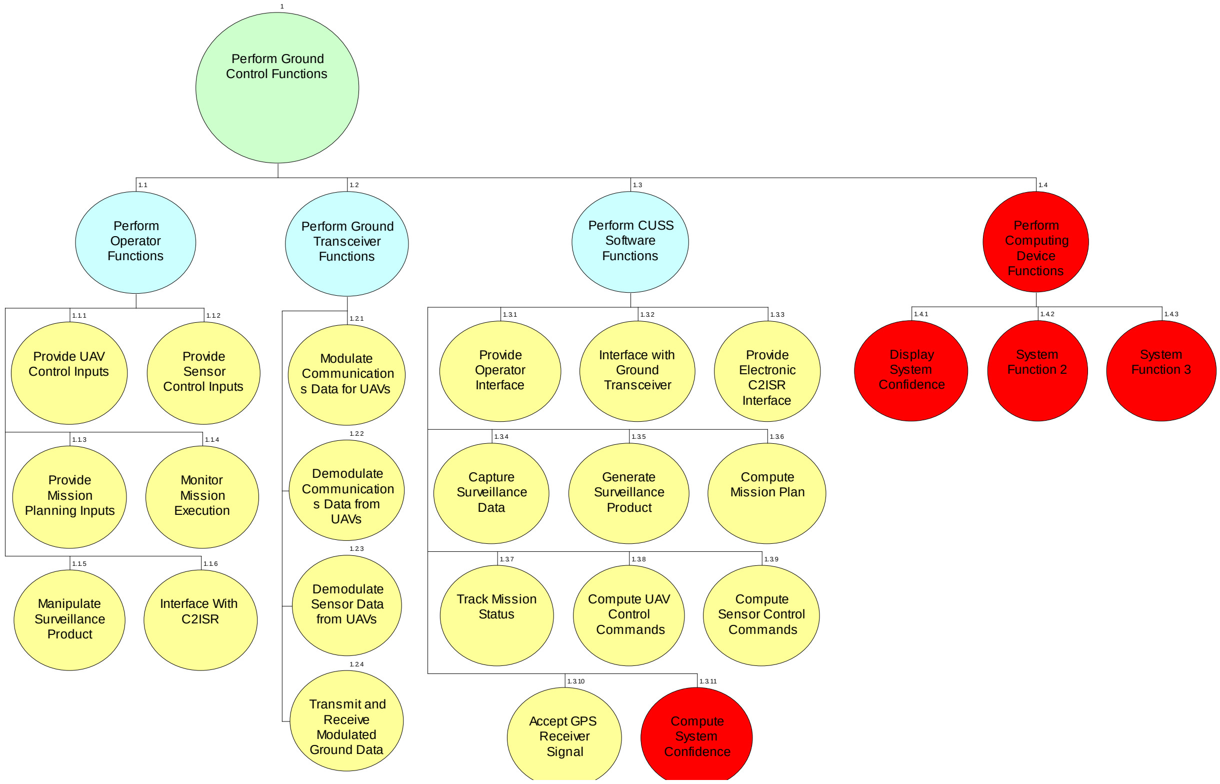

1
Perform Ground Control Functions
1.1
Perform Operator Functions
1.2
Perform Ground Transceiver Functions
1.3
Perform CUSS Software Functions
1.4
Perform Computing Device Functions
1.1.1
Provide UAV Control Inputs
1.1.2
Provide Sensor Control Inputs
1.1.3
Provide Mission Planning Inputs
1.1.4
Monitor Mission Execution
1.1.5
Manipulate Surveillance Product
1.1.6
Interface With C2ISR
1.2.1
Modulate Communications Data for UAVs
1.2.2
Demodulate Communications Data from UAVs
1.2.3
Demodulate Sensor Data from UAVs
1.2.4
Transmit and Receive Modulated Ground Data
1.3.1
Provide Operator Interface
1.3.2
Interface with Ground Transceiver
1.3.3
Provide Electronic C2ISR Interface
1.3.4
Capture Surveillance Data
1.3.5
Generate Surveillance Product
1.3.6
Compute Mission Plan
1.3.7
Track Mission Status
1.3.8
Compute UAV Control Commands
1.3.9
Compute Sensor Control Commands
1.3.10
Accept GPS Receiver Signal
1.3.11
Compute System Confidence
1.4.1
Display System Confidence
1.4.2
System Function 2
1.4.3
System Function 3

Figure 27: SV-4 Diagram with Trust Factors Added

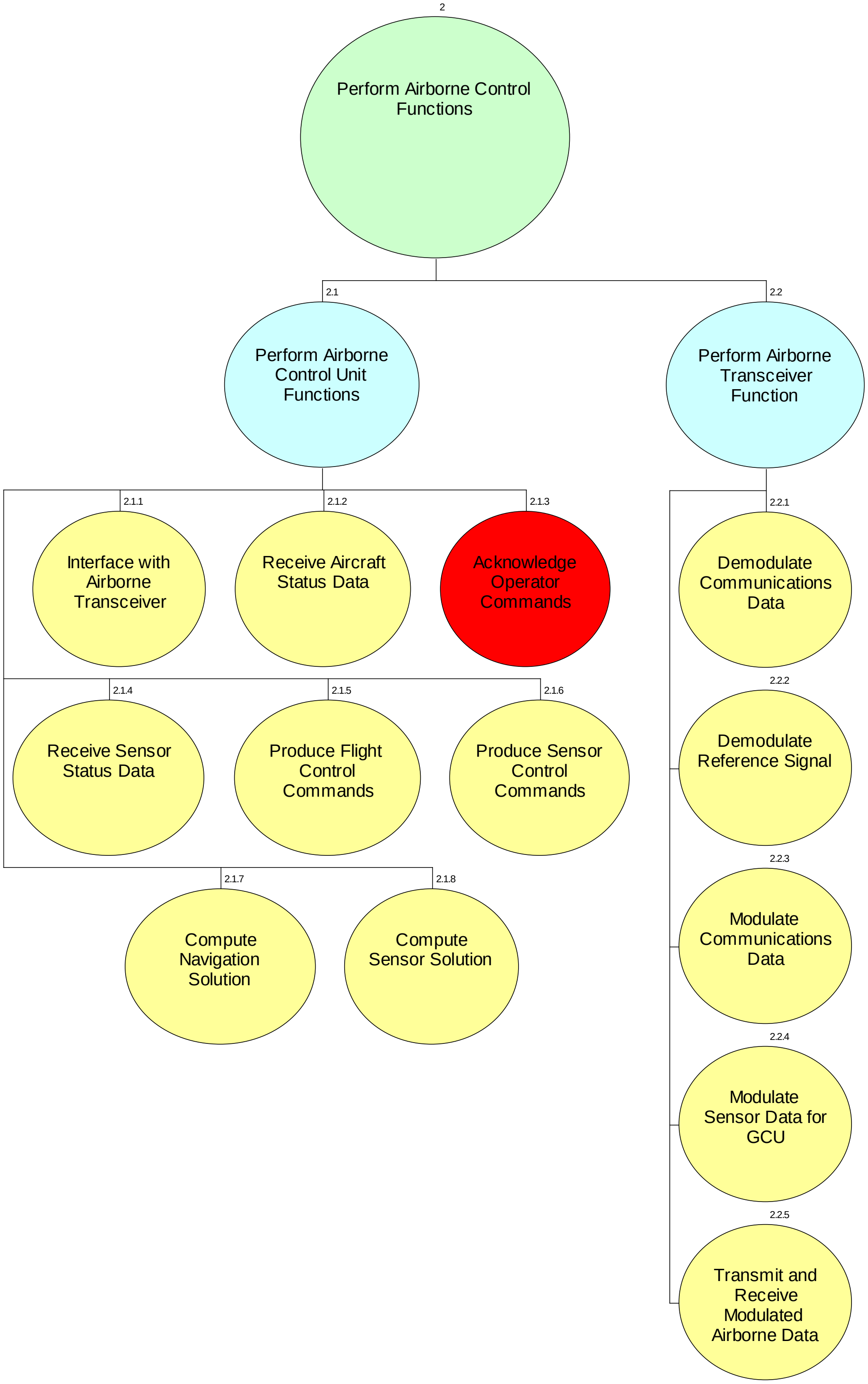

2
Perform Airborne Control Functions
2.1
Perform Airborne Control Unit Functions
2.2
Perform Airborne Transceiver Function
2.1.1
Interface with Airborne Transceiver
2.1.2
Receive Aircraft Status Data
2.1.3
Acknowledge Operator Commands
2.1.4
Receive Sensor Status Data
2.1.5
Produce Flight Control Commands
2.1.6
Produce Sensor Control Commands
2.1.7
Compute Navigation Solution
2.1.8
Compute Sensor Solution
2.2.1
Demodulate Communications Data
2.2.2
Demodulate Reference Signal
2.2.3
Modulate Communications Data
2.2.4
Modulate Sensor Data for GCU
2.2.5
Transmit and Receive Modulated Airborne Data

Figure 28: SV-4 Diagram with Trust Factors Added

4.4.3.2 Informational

The information flows between system components as well as between system automation and the operator are vital in designing for appropriate trust. For example, certain nodes will need to provide detailed information to related nodes in order to calculate and display system confidence to the operator. The OV-7 Logical Data model is a current architecture diagram that details the information flows that occur during operational activities. The diagram below is an example OV-7 for a BATCAM CUSS related system that was developed by AFIT students.

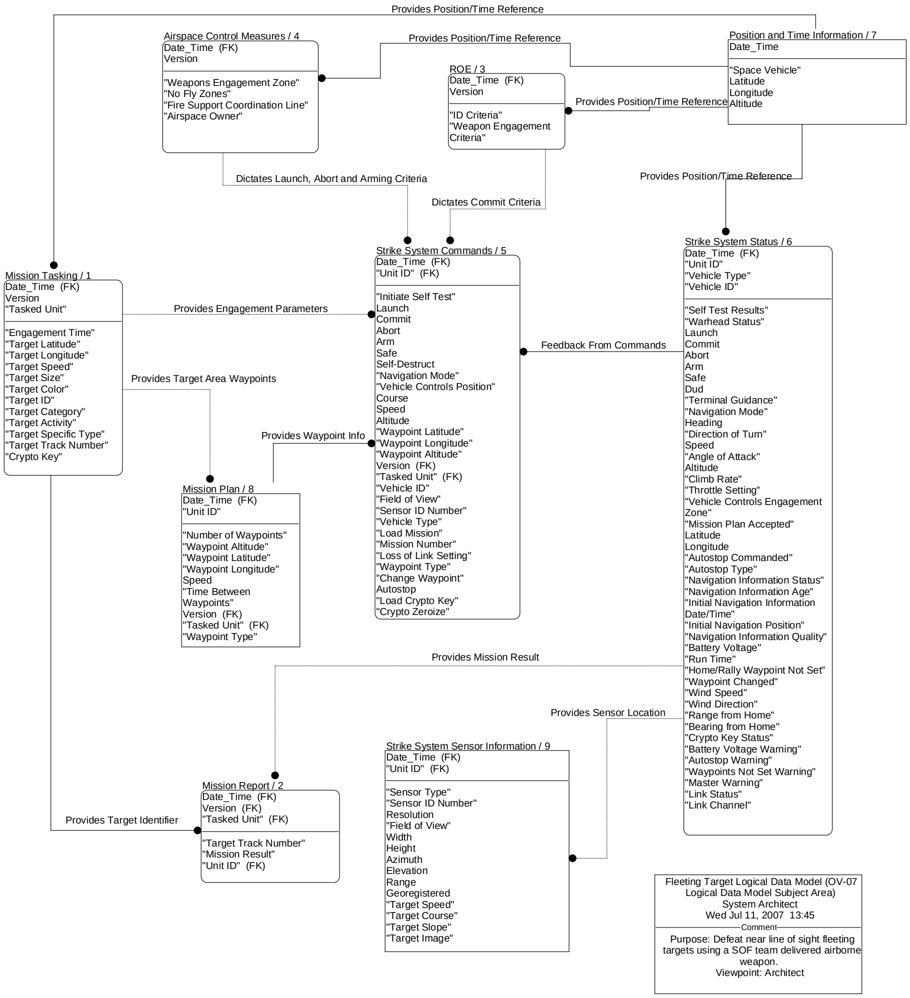


Figure 29: CUSS Related OV-7 Logical Data Model

In order to calculate UAS system confidence, the algorithm that computes the trust factor needs to obtain information from the different nodes that contain the information. For example, the airframe's position and time references are vital components in ensuring that the UAS is flying at the coordinates the operator commanded it to. In addition, environment factors that affect the performance of the aircraft, such as wind speed, provide additional information that may cause the aircraft to deviate from its course. This is especially true in smaller airframes such as the BATCAM UAV. The operators consistently performed visual checks on the aircraft during test flights when the winds were high to ensure it was flying on the preprogrammed path and at the correct altitude. Figure 30 portrays the OV-7 Logical Data Model after the system confidence node was included along with the varying information exchanges that are required to display the measure to the operator.

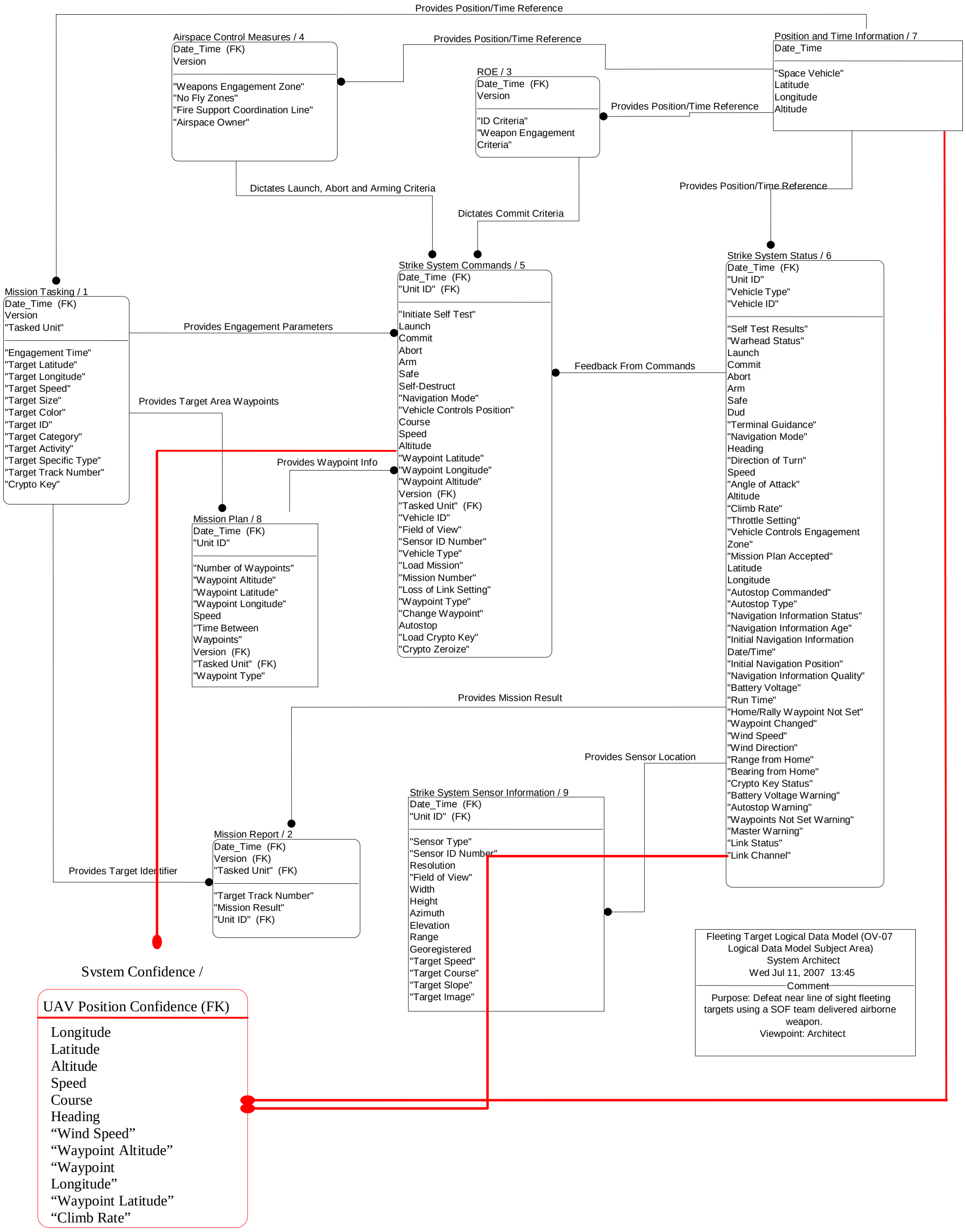


Figure 30: CUSS Related OV-7 Logical Data Model with Trust Factors Included

4.4.3.3 Sequence/Temporal

The temporal architecture views provide an opportunity to examine the tasks that are completed during each scenario and when they are completed with relation to the other tasks.

This breakdown of scenarios provides engineers the information necessary to add the additional tasks outlined in the functional diagrams in the appropriate place to provide pertinent information to the operator at the right time. In addition, the sequence diagrams provide the framework for deciding the type and level of automation that is most effective for each scenario. Figure 31 is an OV-6C Operational Event Diagram depicting the sequence of events that would occur during normal convoy route surveillance or surveillance of a town.

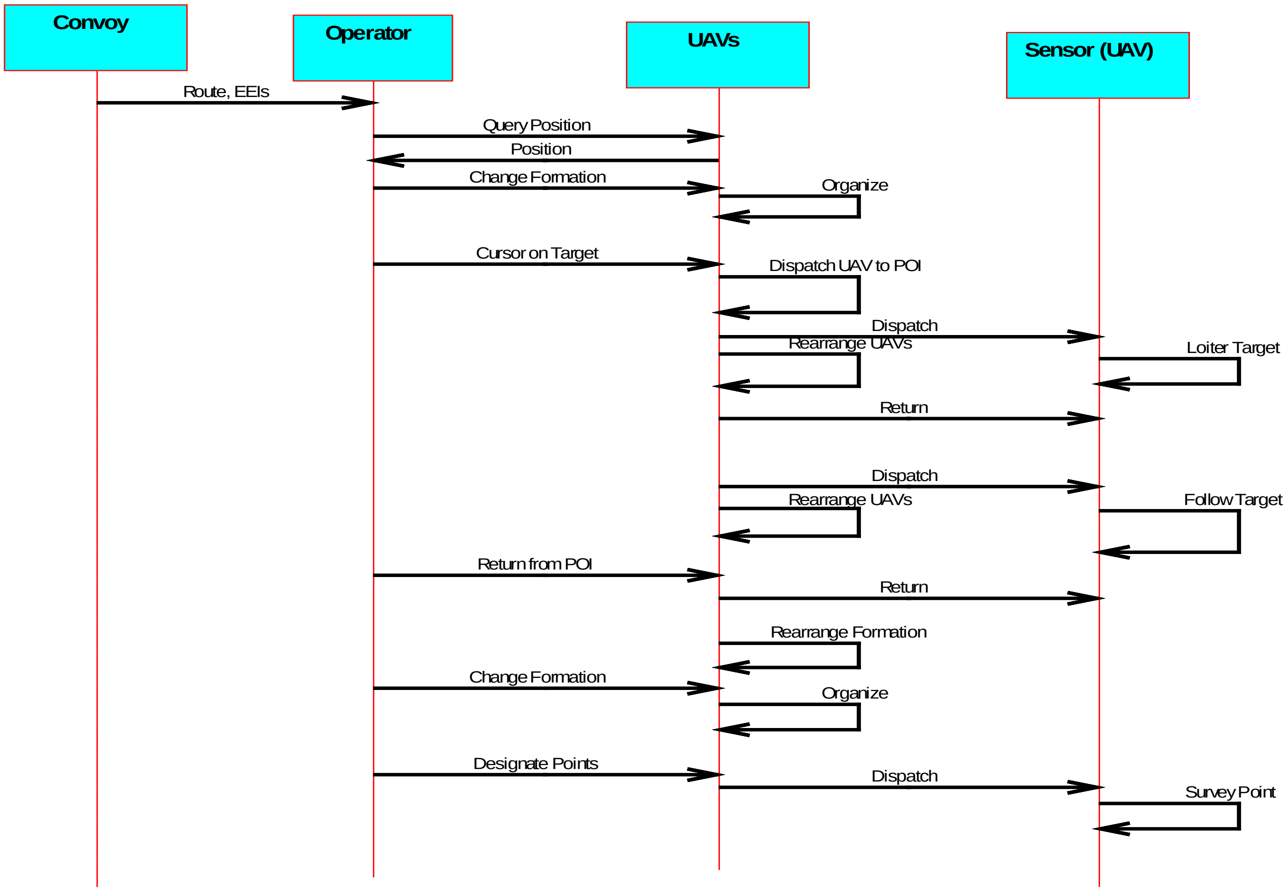


Figure 31: OV-6c Operational Event Diagram for BATCAM Route Surveillance

One of the tasks added in the Functional diagrams designed for increased trust was the acknowledgement of commands from the UAV to the operator. Commands should be acknowledged as soon as they are received in order to allow the operator to form realistic expectations of system behavior. Figure 32 portrays the Operational Event Diagram updated to reflect these acknowledgements.

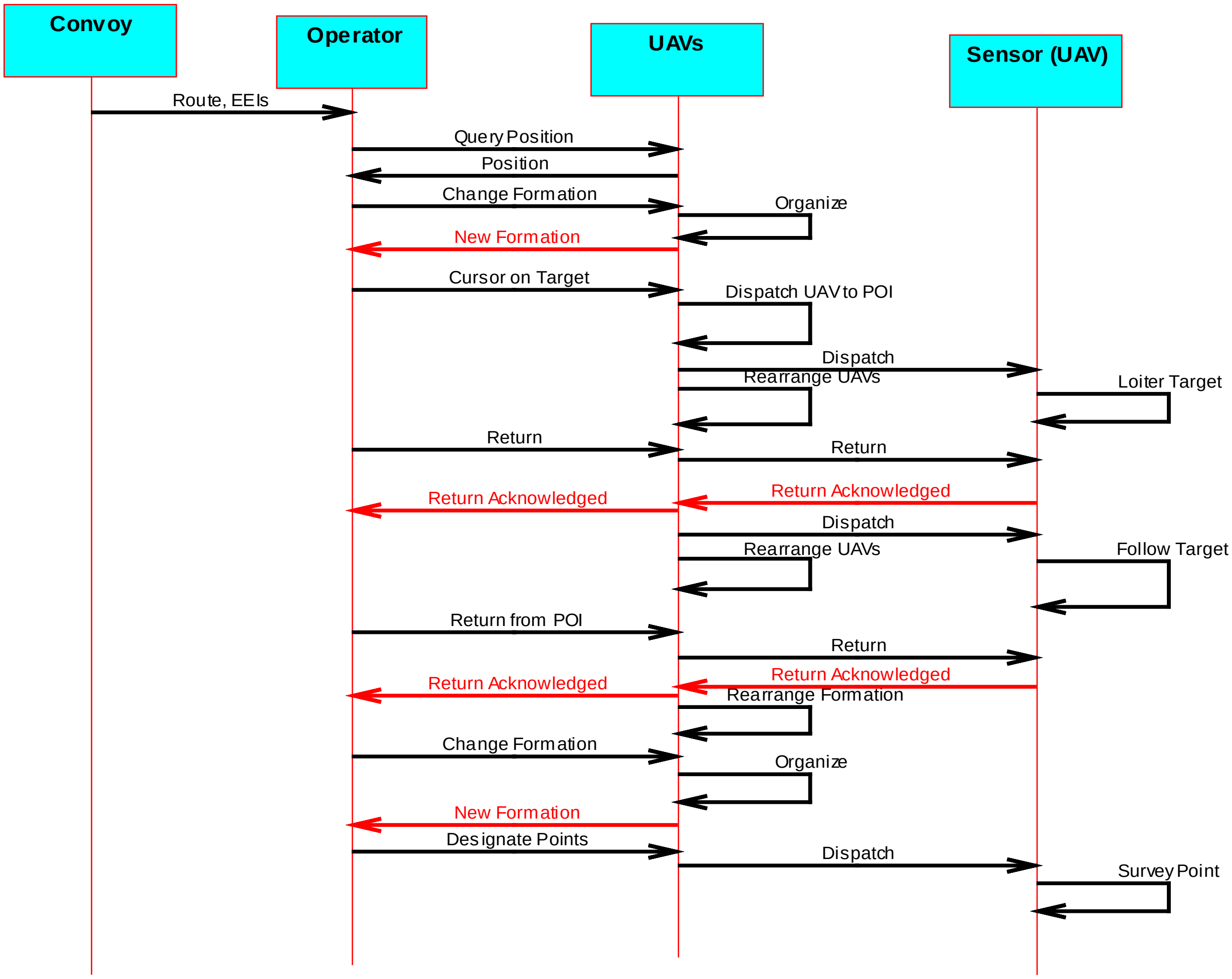


Figure 32: Operational Event Diagram for BATCAM Route Surveillance with Trust Factors Added

As mentioned before, sequence diagrams also help determine the most effective level and type of automation for each scenario. In Figures 31 and 32, the first half of the scenario deals with normal convoy route surveillance. The terrain in this situation is less dynamic and threatening and operators would utilize higher levels of automation to conduct the surveillance. The latter half of the diagram represents the surveillance that would occur as the convoy enters a town. This scenario is more dynamic with potentially higher levels of threat and therefore the operator would need greater SA in order to improve his chances of locating threats in a timely manner. Engineers may want to consider automation that adapts to operator needs and situational conditions, such as that described by Parasuraman, *et al.* (2007), in these scenarios to enhance mission effectiveness.

4.4.3.4 Structural

After an architect is finished applying the trust lens to ensure that appropriate trust dependencies have been included in system design, it is essential to examine the structural layout of the system in order to make certain that the interconnections and interfaces exist between the different subsystems to facilitate system functionality. Figure 33 portrays the BATCAM SV-01 Systems and Services Interfaces Description.

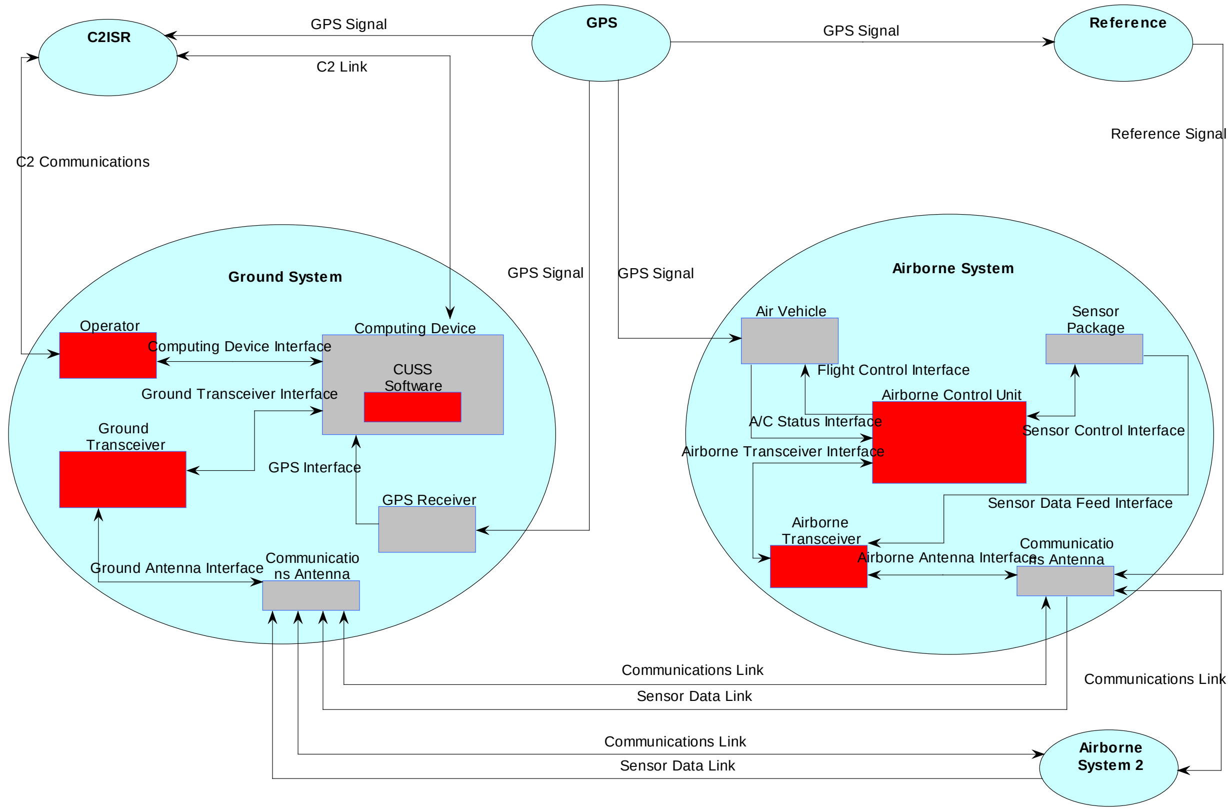


Figure 33 - BATCAM SV-01 System Interface Diagram

First, it is necessary to know where the added trust functions would exist in the structural view. According to Figure 27, the task "acknowledge operator commands" would reside in the Airborne Control Unit, "calculate system confidence" would be in the CUSS Software, and "display system confidence" would be the responsibility of the Computing Device. To accomplish these tasks, the systems that perform the functions must interface with each other. Examining Figure 33, it is apparent the Airborne System interfaces with the Ground System via Comm and Sensor Data Links. Therefore, no additional interfaces need to be implemented in order to facilitate the acknowledgement of operator commands.

Chapter 5: Conclusions and Recommendations

5.1 Conclusions

The increasing complexity of automated systems, and the need to operate them in ever greater numbers with fewer human operators, requires close examination of the relationship between human and machine. This is especially true in the cognitive domain, which is difficult to quantify, model, and measure objectively. Trust is one of the most important aspects of man/machine relationships due to its impact on the human operator's willingness to use the automated system to its full potential. However, operators must also recognize the limitations inherent in the system automation and intervene when the automation's capability has been exceeded. Failure to trust appropriately often results in decreased mission effectiveness and sometimes in catastrophe.

Due to this mis-calibration of trust, systems engineers must understand how trust is formed, maintained, and lost, and account for man/machine trust relationships during system design. This is especially important for distributed systems such as UASs, where operators are not physically present to observe system behavior. The following conclusions from this effort's research questions represent a means to that end.

5.1.1 Can previous research be used to create a useful trust model for systems engineering?

A review of trust-related literature spanning more than fifty years of research enabled the creation of a pedigreed trust model. Although based on established sources by respected researchers, the model itself is an unprecedented attempt to enable the integration of human operator trust within system architectures. This model describes the interaction of context and perception to form expectations, which are the primary inputs to trust. It portrays the attributes (bases) and methods (dynamics) that define trust itself, and the effect of trust on operator decisions (reliance). It depicts reliance as it affects the outcome of actions taken by the operator and/or system. Lastly, it acknowledges the evolving nature of the trust attitude over time with a feedback loop from event outcome(s) to context. The inclusion of social and individual context as inputs to trust also acknowledges its evolving nature, since context derives largely from prior experience. The model is underpinned by the mapping of trust-related factors to the elements of a decision tree representing the reliance decision itself, including an assignment of probability and utility of the possible outcomes of the decision.

Since the model developed here is based on decades of precedent, it is not surprising to note that it bears similarities to segments of previous models (Muir, 1994); (Lee & See, 2004). Specifically, Muir's (1994) model of trust forms the basis for much of the current understanding of trust itself, i.e. its attributes and dynamics. Meanwhile, Lee & See's (2004, p. 68) broader model portrays the influence of context on expectations (shown as belief), trust, and reliance and the direct impact of expectations on trust, and trust on reliance. The model presented here, however, separates perception from context and depicts their association through the formation of expectations which influence trust. Further, this model decomposes perception and context into subclasses known to affect trust. This decomposition enables systems engineers to address each in the most suitable way when designing the system.

So, while the model proposed here is by no means the only “correct” way to portray trust, the model is useful to systems engineers by providing a concrete understanding of trust relationships in a way compatible with the development of system architecture. Thus, the model allows system architects to apply a trust lens to develop automation that is more responsive to specific perceptual and contextual needs of the user. The model also allows architects and other decision makers to address other aspects of the trust relationship through organization, training, etc. The enhanced trust relationship may improve mission effectiveness by decreasing misuse and disuse of automation.

5.1.2 How can trust be considered explicitly within DoDAF?

In order for a trust model to be relevant to systems engineering, it must be applicable to the process of planning and designing systems to achieve required capabilities. The pedigreed model developed in this research is intended as such a resource. The model's representation as a UML class diagram enables direct linkages between portions of the trust model and components of system software and hardware. These linkages allow systems engineers to reinforce realistic trust by shaping operator expectations. For example, improving transparency (e.g. acknowledge commands) while adapting system behavior to account for operator perception (e.g. adaptive automation to affect situational awareness) makes systems more predictable, and thus improves trust.

More importantly, the trust lens, or understanding of trust's impact on system effectiveness fostered by the model, can be applied continuously during the architecture development process. This focus on process over specific products is consistent with the future direction of the DoD Architecture Framework (DoDAF Version 2.0, 2008). Focusing on process also makes the model more broadly useful, since it is not specific to a particular framework or set of standards. This flexibility ensures the model will remain useful as frameworks change or fit-for-purpose views are created.

As with most aspects of HSI, the trust lens is most effective in improving system performance when it is applied from the outset. As discussed in Chapter 4, trust relationships can begin to be evaluated from the moment system requirements begin to be defined. However, as the BATCAM example shows, there is also value in considering trust later in the lifecycle. In short, the application of a trust lens to system architecture can enhance the relationship between humans and automation, and potentially improve system effectiveness.

5.1.3 Can the utility of architecting trust be demonstrated on a given UAS architecture?

By applying a process-oriented approach to architecture development using a trust lens, may be possible to improve the trust relationship between the operator and automation, primarily by increasing transparency and predictability of key functions. As mentioned, while one would expect the trust lens to be most effective when employed from the beginning of architecture development, it might also be applicable to existing systems to improve performance, as demonstrated with the BATCAM example. However, further testing is required to quantify actual improvement to mission effectiveness.

The utility of the trust model, and its use to evaluate architecture through a trust lens, was demonstrated theoretically by considering an existing, though immature, UAS architecture under specific surveillance scenarios. After the addition of trust-related tasks and information exchanges, additional knowledge is available to the operator (e.g. transition between operating modes, system confidence in its ability to operate effectively within a defined envelope, etc). This increased awareness may enable more realistic expectations and thus improve trust. Appropriate trust, in turn, should improve system effectiveness and enable transition from multiple operators on one UAV to one operator multiple UAVs.

This qualitative evaluation, however, is of limited use in determining the full potential of a “trust lens approach” to system architecture. A quantitative evaluation of system effectiveness before and after the inclusion of trust is necessary in order to fully address this research question.

5.2 Recommendations

5.2.1 Develop Comprehensive Test Plan to Quantify Utility of Architecting Trust

One of the main objectives of this thesis was to demonstrate the utility of architecting trust on a given UAS architecture. However, a thorough test and evaluation was not accomplished due to time constraints and the immaturity of AFIT's multi-UAV implementation of BATCAM system. Therefore, the research team recommends a follow-on study to develop a comprehensive test plan including detailed measures of performance, effectiveness, and suitability, to examine how the system performs before and after trust relationships are integrated into the architecture. The test can be either conducted on the BATCAM UAS when it reaches a state of maturity that will allow sufficient test data to be collected, a similar single operator multi-UAV system, or any operationally relevant system with complex operator-automation interaction. The collection and analysis of test data will ultimately prove whether mission effectiveness is enhanced by engineering for appropriate trust.

5.2.2 Extend Trust Model to Show Team Collaboration

The notional trust model that was developed in Chapter 3 represents one operator interacting with a particular system. However, most military engagements are never this simple. In real-world systems, multiple operators interact with one another and a variety of systems. Therefore, to extend the utility of this research, it would be beneficial to extend the trust model to incorporate team collaboration and the interdependencies of trust between different operators. A researcher may be able to use the Distributed Common Ground Systems (DCGS) or similar team environment, for the basis of their study of the trust interactions, interdependencies, and relationships in a collaborative atmosphere.

5.2.3 Perform HFE Study to Determine Most Effective Means to Convey Trust Data

Architecting for trust alone will not necessarily lead an operator to more appropriately rely on automation. Instead, trust related data depicted in the architecture must be properly conveyed to the operator. Jamieson, *et al.* (2008) conducted experiments and developed several display prototypes that represented trust factor information (i.e. displaying system confidence) to the user in different ways. The data they collected during the experiment indicated that the manner in which the data was presented affected the participants' reliance upon the system (Jamieson, Wang, & Neyedli, 2008). Therefore, the system interface design should account for these factors and displays trust factors in a manner which induces appropriate operator reliance upon the system. This research group recommends a human factors engineering (HFE) study of how to appropriately convey trust-related data to UAV operators (e.g. visual, auditory, size/color of display, etc.).

5.2.4 Develop State Transition Diagram for Evolution of Human/Machine Trust

The main goal of this research effort was to incorporate trust into system architecture in order to foster appropriate trust. As operators interact with systems repeatedly over time, the trust relationship should evolve to where the user has appropriate trust in the system and is not misusing or disusing the system. The goal in applying a trust lens is to minimize this evolution time and to drive the operator to the appropriate trust state. Therefore, the authors recommend a study to develop a state transition diagram or similar temporal flow diagram that describes the evolution of operator/automation trust over time for a specific system and/or set of scenarios.

5.2.5 Determine Appropriate Degree of Transparency

The authors of this research have advocated designing systems with transparency of automation to help facilitate appropriate trust. However, the scope of this research did not allow determination of the most effective level or type of transparency. For example, if transparency is insufficient operators may not trust the system to produce a desired outcome. Conversely, too much transparency of system behavior has the potential to overwhelm the operator with unnecessary information. Previous research efforts have addressed similar concerns; for example, Cummings, *et al.* has measured the effects of different levels of automation on situational awareness and reliance (Cummings & Mitchell, 2008); (Cummings, Bruni, Mercier, & Mitchell, 2007). However, the authors are not aware of any research that has attempted to balance the trust benefits of transparency with the limited processing ability of human operators. Therefore, the authors recommend a study to determine the degree of transparency that will best enhance mission effectiveness for a given system.

5.2.6 Develop a Trust Profile

Reliance is a decision based largely on the trust attitude (Lee & See, Trust in Automation: Designing for Appropriate Reliance, 2004). Prior studies have proposed methods to measure trust empirically (Jian, Bisantz, & Drury, 2000), yet have not attempted to predict differences in behavior based on the measured degree of trust. Expected utility theory, meanwhile, contends that individuals tend to make different decisions in the same situation depending on their risk attitude (e.g. risk seeking, risk neutral, or risk averse.) Since the need to trust is explicitly tied to the presence of uncertainty and risk in a given situation (Bhattacharya, Devinney, & Pillutla, 1998), it follows that an individual's inclination to trust may bear some relation to their risk attitude as described in utility theory, and hence affects their behavior.

Bhattacharya, *et al.*'s (1998) research on trust relates to utility theory in its attention to risk, but could be extended by developing a trust profile similar to that for risk in utility theory. In addition, a model of "trust patterns" then could be developed and applied to identify the tendencies of individuals with different trust patterns, and modify the behavior of automation (e.g. amount and type of information presented to a specific operator) in order to elicit appropriate trust from specific individuals in response. Adjusting for individual inclination to trust will negate one of the limitations identified during the development of the trust model discussed in Chapter 3. If applied correctly with the present trust model during system design, engineers may be better able to induce appropriate trust regardless of individual inclination.

Works Cited

Antifakos, S., Kern, N., Schiele, B., & Schwaninger, A. (2005). Towards Improving Trust in Context-Aware Systems by Displaying System Confidence. *ACM International Conference Proceeding Series. 111*, pp. 9-14. Salzburg, Austria: Association for Computing Machinery, New York, NY, USA.

Barber, B. (1983). *The Logic and Limits of Trust.* New Brunswick, NJ: Rutgers University Press.

Bhattacharya, R., Devinney, T. M., & Pillutla, M. M. (1998). A Formal Model of Trust Based on Outcomes. *The Academy of Management Review , 23* (3), 459-472.

Bisantz, A. M., & Seong, Y. (2001). Assessment of operator trust in and utilization of automated decision-aids under different framing conditions. *International Journal of Industrial Ergonomics, 28* (2), 85-97.

Boyd, J. (1987). A discourse on winning and losing. *Air University Library Document No. M-U 43947* . Maxwell Air Force Base, AL.

Cohen, M. S., Parasuraman, R., & Freeman, J. T. (1998). Trust in Decision Aids: A Model and its Training Implications. *Proceedings 1998 Command and Control Research and Technology Symposium*, (pp. 1-37). Monterey, CA.

Committee on Human-System Design Support for Changing Technology. (2007). *Human-System Integration in the System Development Process.* (R. W. Pew, & A. S. Mavor, Eds.) Washington, D.C.: The National Academies Press.

Cummings, M. L., & Mitchell, P. J. (2008). Predicting Controller Capacity in Supervisory Control of Multiple UAVs. *IEEE Transactions on Systems, Man, and Cybernetics , 38* (2), 451-460.

Cummings, M. L., Bruni, S., Mercier, S., & Mitchell, P. J. (2007). Automation Architecture for Single Operator, Multiple UAV Command and Control. (S. Mulgund, Ed.) *The International C2 Journal , 1* (2), 1-24.

Dictionary.com. (2006). Retrieved January 7, 2009, from Dictionary.com: http://dictionary.reference.com/

DoDAF Version 1.5. (2007, April 27). *DoDAF Volume II.* Retrieved December 31, 2008, from DefenseLink: http://www.defenselink.mil/cio-nii/docs/DoDAF_Volume_II.pdf

DoDAF Version 2.0. (2008, December). *DRAFT DoDAF v2.0 Volume 1*, DoDAF v2.0. (U.S. Department of Defense) Retrieved December 31, 2008

Drury, J., & Scott, S. (2008). Awareness in Unmanned Aerial Vehicle Operations. *The International C2 Journal* .

Dzindolet, M. T., Peterson, S. A., Pomranky, R. A., Pierce, L. G., & Beck, H. P. (2003). The Role of Trust in Automation Reliance. (S. Wiedenbeck, B. Kracher, & C. Corritore, Eds.) *International Journal of Human-Computer Studies , 58* (6), 697-718.

Dzindolet, M. T., Pierce, L. G., Beck, H. P., Dawe, L. A., & Anderson, B. W. (2001). Predicting Misuse and Disuse of Combat Identification Systems. *Military Psychology , 13* (3), 147-164.

Endsley, M. R., & Garland, D. J. (2000). *Situation Awareness: Analysis and Measurement* (Illustrated ed.). Lawrence Erlbaum Associates.

Fitts, P. M., & Posner, M. I. (1967). *Human Performance.* Belmont, CA: Brooks/Cole Pub. Co.

Gao, J., & Lee, J. D. (2006). Extending the Decision Field Theory to Model Operators' Reliance on Automation in Supervisory Control Situations. *IEEE transactions on systems, man and cybernetics. Part A, Systems and humans , 36* (5), 943-959.

Halpin, S., Johnson, E., & Thornberry, J. (1973). Cognitive Reliability in Manned Systems. *IEEE Transactions on Reliability , 22*, 165-169.

INCOSE. (August 2007). INCOSE Systems Engineering Handbook v3.1, Appendix M - Human Systems Integration.

Jamieson, G. A., Wang, L., & Neyedli, H. F. (2008). *Developing Human-Machine Interfaces to Support Appropriate Trust and Reliance on Automated Combat Identification Systems.* Toronto: University of Toronto.

Jian, J.-Y., Bisantz, A. M., & Drury, C. G. (2000). Foundations for an Empirically Determined Scale of Trust in Automated Systems. *International Journal of Cognitive Ergonomics , 4* (1), 53-71.

Joint Chiefs of Staff. (2007, May). Joint Capabilities and Integration Development System. *CJCSI 3170.01F* .

Joint Pub 1-02. (2001, April 12). *Department of Defense Dictionary of Military and Associated Terms.* Retrieved January 11, 2009, from Joint Doctrine, Education and Training Electronic Information System: http://www.dtic.mil/doctrine/jel/new_pubs/jp1_02.pdf

Konstantinou, E., Liagkou, V., Spirakis, P., Stamatiou, Y., & Yung, M. (2005). "Trust Engineering:" From Requirements to System Design and Maintenance - A Working National Lottery System Experience. *8th International Conference, ISC 2005, Singapore, September 20-23, 2005. Proceedings. 3650/2005*, pp. 44-58. Singapore: Springer Berlin / Heidelberg.

Lee, J. D., & Moray, N. (1994). Trust, self-confidence, and operators' adaptation to automation. *International Journal of Human-Computer Studies , 40* (1), 153-184.

Lee, J. D., & See, K. A. (2004). Trust in Automation: Designing for Appropriate Reliance. (N. J. Cooke, Ed.) *Human Factors: The Journal of the Human Factors and Ergonomics Society , 46* (1), 50-80.

Miller, J. E. (2008). Uncovering Expectations to Support Trust in System Development. *World Congress in Computer Science Computer Engineering and Applied Computing.* Las Vegas.

Moray, N., & Inagaki, T. (2000). Attention and complacency. *Theoretical Issues in Ergonomics ,* 354–365.

Muir, B. M. (1987). Trust between humans and machines, and the design of decision aids. *International Journal of Man-Machine Studies , 27* (5-6), 527-539.

Muir, B. M. (1994). Trust in Automation: Part I. Theoretical issues in the study of trust and human intervention in automated systems. *Ergonomics , 37* (11), 1905-1922.

Parasuraman, R., & Riley, V. (1997). Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors: The Journal of the Human Factors and Ergonomics Society , 39* (2), 230-253.

Parasuraman, R., Barnes, M., & Cosenzo, K. (2007). Adaptive Automation for Human-Robot Teaming in Future Command and Control Systems. (S. Mulgund, Ed.) *The International C2 Journal , 1* (2), 43-68.

Parasuraman, R., Sheridan, T. B., & Wickens, C. D. (2008). Situation Awareness, Mental Workload, and Trust in Automation: Viable, Empirically Supported Cognitive Engineering Constructs. *Journal of Cognitive Engineering and Decision Making , 2* (2), 140-160.

Parasuraman, R., Sheridan, T., & Wickens, C. (2000). A model for types and levels of human interaction with automation. *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans , 30* (3), 286-297.

Parsch, A. (2005). *Appendix 4: Undesignated Vehicles*. Retrieved March 8, 2009, from Directory of U.S. Military Rockets and Missiles: http://www.designation-systems.net/dusrm/app4/batcam.html

Pina, P. E., Donmez, B., & Cummings, M. L. (2008, May). *Selecting Metrics to Evaluate Human Supervisory.* Retrieved November 24, 2008, from MIT Humans and Automation Lab: http://web.mit.edu/aeroastro/labs/halab/papers/Pina_Metrics.pdf

Procerus Technologies. (2006). *Procerus Technologies: Kestrel Autopilot*. Retrieved March 8, 2009, from Procerus Technologies: http://www.procerusuav.com/productsKestrelAutopilot.php

Rempel, J. K., Holmes, J. G., & Zanna, M. P. (1985). Trust in close relationships. *Journal of Personality and Social Psychology , 49* (1), 95-112.

Sakryd, G. A., & Ericson, D. A. (2008, March). Systems Engineering Analysis for the Development of the Fleeting Target Technology Demonstrator. *Masters Thesis* . WPAFB, OH: Air Force Institute of Technology.

Senders, J. (1964). The human operator as a monitor and controller of multidegree of freedom. *IEEE Transactions on Human Factors in Electronics* , 2-6.

Sheridan, T. B. (1988). Trustworthiness of command and control systems. *Proceedings of IFAC Man–Machine Systems*, (pp. 427-431). Oulu, Finland.

Wikipedia contributors . (2009, January 7). *Wikipedia*, 262476840. Retrieved January 15, 2009, from Wikipedia: http://en.wikipedia.org/w/index.php?title=Qualia&oldid=26247684

Wikipedia contributors. (2009, January 9). *Wikipedia*, 262878787. Retrieved January 22, 2009, from Wikipedia: http://en.wikipedia.org/w/index.php?title=Expected_utility_hypothesis&oldid=262878787

Williams, K. W. (2006). *Human Factors Implications of Unmanned Aircraft Accidents: Flight-Control Problems.* Oklahoma City: Civil Aerospace Medical Institute.

Zall, J., & O'Malley, D. (2008). Architecting Cognition within the Department of Defense Architecture Framework.

Zuboff, S. (1988). *In the age of smart machines: The future of work technology and power*. New York: Basic Books.

Appendix A – Trust Lens Implementation Guide

While the research presented here was inspired by the need for improved UAV automation, the authors believe the “trust lens” can be applied to improve the effectiveness of any system in which humans interact with complex automation. Thus, systems engineers, program managers, and the like may wonder how they can apply the trust model presented here to their specific program. The following quick reference guide is provided as a starting point for considering trust during system architecture development. It relates the trust model elements most immediately relevant for consideration during each architecture development step. Keep in mind that the architecture development steps are iterative, not sequential.

Steps 1 & 2: Determine Scope and View

- Identify human/machine relationships and dependencies
- Plan for impact of context on the operator’s trust in the system (see sections 2.2.1.1 and 3.2.2)
    - Consider existing/desired social and organizational context (see section 2.2.2)
    - Begin to shape individual context by identifying manpower, personnel, and training (MPT) requirements (see sections 2.5 and 3.2.2)

Steps 3 & 4: Determine Data Required & Collect, Organize, Correlate and Store the Data

- Use scenario data (e.g. sequence diagrams) to model and predict operator perception (e.g. SA, time stress) under expected conditions (see sections 2.2.1.5, 3.2.1, 4.3.2.3 and 4.4.3.3)
- Design automation to elicit realistic expectations by providing appropriate transparency and predictability of system state (see sections 2.2.2, 3.2.2 and 3.2.3)

Steps 5 & 6: Conduct Analysis and Document Results

- Validate or refine MPT requirements that affect context, scenarios that affect operator perception models, and design requirements to improve automation transparency and predictability
- Build architecture, incorporating tasks and information flows identified in steps 3 & 4
- Refine and finalize training objectives and scenarios to represent system behavior in operationally realistic conditions

Appendix B – Glossary

Ability: the group of skills, competencies, and characteristics that enable the trustee to influence the domain. (Lee & See, 2004)

Adaptive Automation: automation that adapts to the needs of the operator in a given situation by altering its behavior, e.g. shifts from fully automated to compliance-oriented to minimal decision support, when the system reaches predetermined states

Appropriateness of trust: the relationship between the true capabilities of the agent and the level of trust (Lee & See, 2004)

Attitude: an affective evaluation of beliefs that guides people to adopt a particular intention. (Lee & See, 2004)

Automation: the technology that actively selects data, transforms information, makes decisions, or controls processes. (Lee & See, 2004)

Automation bias: a decision bias that occurs when operators become over-reliant on the automation and do not verify the accuracy of automated recommendations. (Cummings, Bruni, Mercier, & Mitchell, 2007)

Battlespace: The environment, factors, and conditions that must be understood to successfully apply combat power, protect the force, or complete the mission. (Joint Pub 1-02, 2001)

Belief: any cognitive content held as true (Dictionary.com, 2006)

Benevolence: the extent to which the intents and motivations of the trustee are aligned with those of the trustor. (Lee & See, 2004)

Calibration: the correspondence between a person's trust in the automation and the automation's capabilities. (Lee & Moray, Trust, self-confidence, and operators' adaptation to automation, 1994) (Muir, 1987)

Capability: the ability to execute a specified course of action. (A capability may or may not be accompanied by an intention.) (Joint Pub 1-02, 2001)

Complacency: overreliance resulting in failure to monitor the "raw" information sources that provide input to the automated system. (Parasuraman, Sheridan, & Wickens, Situation Awareness, Mental Workload, and Trust in Automation: Viable, Empirically Supported Cognitive Engineering Constructs, 2008)

Compliance-oriented automation: automation indicates an abnormal situation, and the operators act in response to this indicator. (Lee & See, 2004)

Confidence: a level of expectation associated with a particular prediction or belief.

Context, individual: a specific person's history of interactions with the trustee, as well as their general inclination to trust, which can vary from one individual to the next. (Lee & See, 2004)

Context, organizational: interactions between people that inform them about the trustworthiness of others, which can include reputation and gossip. (Lee & See, 2004)

Context, situational: interactions between people and the environment and current state of the system

Context, social: a set of social norms and expectations that influence trust through shared educational and life experiences associated with cultural differences or distinct groups of workers.

Controls: the rules, doctrine, regulations, or other documents that prescribe how an action is to take place, what course the activity must follow, and, what form or format is expected/required for the result. (DoDAF Version 2.0, 2008)

Dependability: the extent to which the trustee can be relied upon. (Muir, 1994)

Disuse: failures that occur when people reject the capabilities of automation. (Lee & See, 2004)

Effectiveness: adequate to accomplish a purpose; producing the intended or expected result (Dictionary.com, 2006)

Environment: the aggregate of surrounding things, conditions, or influences; surroundings; milieu. (Dictionary.com, 2006)

Expectation: belief about (or mental picture of) the future (Dictionary.com, 2006)

Expected Utility Theorem: predicts that the "betting preferences" of people with regard to uncertain outcomes can be described by a mathematical relation which takes into account the size of a payout, the probability of occurrence, risk aversion, and the different utility of the same payout to people with different assets or personal preferences. (Wikipedia contributors, 2009)

Experience: knowledge or practical wisdom gained from what one has observed, encountered, or undergone (Dictionary.com, 2006)

Faith: a judgment that the trustee can be relied on, even though specific actions and their outcomes may be unknown at the time. Also, a closure against doubt in the face of an uncertain future. (Muir, 1994)

Fiduciary Responsibility: an assumption that the trustee will act appropriately when trustee competence is unknown. e.g. An operator assumes a system will meet its design-based performance criteria when it is operating as intended. (Muir, 1994)

General Inclination: the inherent tendency a specific individual has to place trust in an unknown system

Human System Integration: The interdisciplinary technical and management processes for integrating human considerations within and across all system elements; an essential enabler to systems engineering practice (INCOSE Systems Engineering Handbook)

Importance: the value of the expected outcome to the trustor. (Bhattacharya, Devinney, & Pillutla, 1998)

Information element: a formalized representation of information subject to an operational process (e.g., the information content that is required to be exchanged between nodes). (DoDAF Version 1.5, 2007)

Information exchange: an act of exchanging information between two distinct operational nodes. Characteristics of the act include the information element that needs to be exchanged and its attributes (e.g., Scope), as well as attributes associated with the exchange (e.g., Transaction Type). (DoDAF Version 1.5, 2007)

Inputs: the triggers that cause an activity to occur, along with data or information needed to perform the desired action. (DoDAF Version 2.0, 2008)

Integrity: the degree to which the trustee adheres to a set of principles the trustor finds acceptable. (Lee & See, 2004)

Interaction Time: the time it takes for a human to interact with the automation to raise the performance to an acceptable level. (Cummings & Mitchell, 2008)

Interface: a common boundary or interconnection between systems, equipment, concepts, or human beings. (Dictionary.com, 2006)

Mechanisms: see Resources.

Mental Workload: the relation between the function relating the mental resources by a task and those resources available to be supplied by the human operator. (Parasuraman, Sheridan, & Wickens, Situation Awareness, Mental Workload, and Trust in Automation: Viable, Empirically Supported Cognitive Engineering Constructs, 2008)

Misuse: failures that occur when people inadvertently violate critical assumptions and rely on automation inappropriately. (Lee & See, 2004)

Operator Perception: an operator's intuitive or mental representation of certain factors that affect expectations and attitudes

Outcome: An end result; a consequence (Dictionary.com, 2006)

Outputs: the results of activity performance. These can be outputs of products, services, or requirements for further action, or outcomes (i.e. demonstration that an action has produced a desired change.) (DoDAF Version 2.0, 2008)

Performance: the competency or expertise as demonstrated by its ability to achieve the operator's goals. (Lee & See, 2004)

Persistence: an expectation of constancy, which, for example, allows the operator to construct a mental model of the automation through experience interacting with it. (Muir, 1994)

Predictability: visibility of specific behaviors, or performance, that allow operators to judge automation based on its ability to deliver consistent and desirable results.

Process: the degree to which the automation's algorithms are appropriate for the situation and able to achieve the operator's goals. (Lee & See, 2004)

Purpose: the degree to which the automation is being used within the realm of the designer's intent. (Lee & See, 2004)

Quale: a term used in philosophy to describe the subjective quality of conscious experience. (Wikipedia contributors , 2009)

Queuing Time: the amount of time a decision is prolonged due to the presence of multiple tasks.

Reliance: a behavior representing the operator's intention or willingness to act, in the form of a decision to use or not to use automation.

Resolution: how precisely a judgment of trust differentiates levels of automation capability. (Cohen, Parasuraman, & Freeman, 1998)

Resources: those things that assist in performance of the activity. These can be physical, logical, technological, or human resources. (DoDAF Version 2.0, 2008)

Risk: exposure to the chance of injury or loss. (Dictionary.com, 2006)

Self-Confidence: confidence in one's own judgment, ability, power, etc. (Dictionary.com, 2006)

Situational Awareness: perception of the elements in the environment within a volume of time and space, the comprehension of their meaning, and the projection of their status in the near future. (Parasuraman, Sheridan, & Wickens, Situation Awareness, Mental Workload, and Trust in Automation: Viable, Empirically Supported Cognitive Engineering Constructs, 2008)

Specificity: the degree to which trust is associated with a particular component or aspect of the trustee. (Lee & See, 2004)

Specificity, functional: the differentiation of functions, subfunctions, and modes of automation. e.g. With high functional specificity, a person's trust reflects capabilities of specific subfunctions and modes. Low functional specificity means the person's trust reflects the capabilities of the entire system. (Lee & See, 2004)

Specificity, temporal: changes in trust as a function of the situation or over time. e.g. High temporal specificity means that a person's trust reflects moment-to-moment fluctuations in automation capability, whereas low temporal specificity means that the trust reflects only long-term changes in automation capability. (Lee & See, 2004)

Stakes: the consequences of a mistake, or, similar to importance, the value to the operator of achieving a desired outcome.

Strength: some degree of confidence in the expected outcome. (Bhattacharya, Devinney, & Pillutla, 1998)

Suitability: the quality of having the properties that are right for a specific purpose (Dictionary.com, 2006)

System nodes: containers for service software items along with the corresponding infrastructure software items and physical computing resource items that enable their existence. A system node may contain one or many service, software, and hardware items. (DoDAF Version 1.5, 2007)

System State: the dynamic behaviors concerning the timing and sequencing of events that capture system performance characteristics of an executing system (DoDAF Version 1.5, 2007)

Technical Competence: the trustor believes the trustee has the ability to perform the assigned task successfully. (Muir, 1994)

Time Stress: perceived cost of delay, e.g. when cost of delay is great, action is more imperative, even with high uncertainty about trust. (Cohen, Parasuraman, & Freeman, 1998)

Training: to coach in or accustom to a mode of behavior or performance (Dictionary.com, 2006)

Transparency: the full, accurate, and timely disclosure of information (Dictionary.com, 2006)

Trust: the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability. (Lee & See, 2004)

Trust Lens: an evaluation of systems architectures by a systems engineer, human factors engineer, or other professional to plan for appropriate trust relationships

Uncertainty: consequences of alternative courses of action are not always known or predictable. (Bhattacharya, Devinney, & Pillutla, 1998)

Unpredictability, apparent: inability to predict behavior due to interaction with an unstable environment, rather than the nature of the system itself.

Unpredictability, inherent: inability to predict behavior due to the nature of the system or automation.

Use: refers to the situations where an operators places appropriate trust in the system; i.e. does not misuse or disuse the automation.

Wait Time: the expected amount of time a task or decision will delayed before an operator addresses it.

REPORT DOCUMENTATION PAGE	*Form Approved* *OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)* 26-03-2009	**2. REPORT TYPE** Master's Thesis	**3. DATES COVERED** *(From – To)* June 2008 – Mar 2009

TITLE AND SUBTITLE Architecting Human Operator Trust in Automation to Improve System Effectiveness in Multiple Unmanned Aerial Vehicles (UAV)	5a. CONTRACT NUMBER
	5b. GRANT NUMBER
	5c. PROGRAM ELEMENT NUMBER
AUTHOR(S) Lenfestey, Adam, G., Maj, USAF Cring, Eric, A., Capt, USAF	**5d. PROJECT NUMBER** JON ENV 09-165
	5e. TASK NUMBER
	5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT/GSE/ENV/09-M06
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Dr. Janet Miller Air Force Research Laboratory 2698 G Street, B190 WPAFB, OH 45433-7604 937-656-4401	**10. SPONSOR/MONITOR'S ACRONYM(S)** 711 HPW/RHXB
	11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for Public Release; Distribution Unlimited

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Current Unmanned Aerial System (UAS) designs require multiple operators for each vehicle, partly due to imperfect automation matched with the complex operational environment. This study examines the effectiveness of future UAS automation by explicitly addressing the human/machine trust relationship during system architecting. A pedigreed engineering model of trust between human and machine was developed and applied to a laboratory-developed micro-UAS for Special Operations. This unprecedented investigation answered three primary questions: Can previous research be used to create a useful trust model for systems engineering? How can trust be considered explicitly within the DoD Architecture Framework? Can the utility of architecting trust be demonstrated on a given UAS architecture? By addressing operator trust explicitly during architecture development, system designers can incorporate more effective automation. The results provide the Systems Engineering community a new modeling technique for early human systems integration.

15. SUBJECT TERMS
Trust in Automation, multiple UAV control

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Dr. John Colombi AFIT/ENV
REPORT U	ABSTRACT U	c. THIS PAGE U	UU	112	**19b. TELEPHONE NUMBER** *(Include area code)* 937-785-3636 John.colombi@afit.edu

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. Z39-18

	Form Approved *OMB No. 074-0188*