AFRL-RH-WP-TP-2010-0002

Eliciting Expectations to Develop Trust in Systems

Janet E. Miller
Cheryl Batchelor

Air Force Research Laboratory
Human Effectiveness Directorate

LeeAnn Perkins

Advanced Virtual Engine Test Cell (AVETEC), Inc.
Springfield OH 45505

May 2008

Interim Report

Air Force Research Laboratory
Human Effectiveness Directorate
Warfighter Interface Division
Cognitive Systems Branch
Wright-Patterson AFB OH 45433-7022

REPORT DOCUMENTATION PAGE			*Form Approved* *OMB No. 0704-0188*

1. REPORT DATE *(DD-MM-YYYY)* May 2008	**2. REPORT TYPE** Interim	**3. DATES COVERED** *(From - To)*
4.TITLE AND SUBTITLE Eliciting Expectations to Develop Trust in Systems		**5a. CONTRACT NUMBER** In-House
		5b. GRANT NUMBER
		5c. PROGRAM ELEMENT NUMBER 63231F
6. AUTHOR(S) [1]Janet E. Miller, [1]Cheryl Batchelor, [2]LeeAnn Perkins		**5d. PROJECT NUMBER**
		5e. TASK NUMBER
		5f. WORK UNIT NUMBER 2830HEX1
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) [2]Advanced Virtual Engine Test Cell (AVETEC), Inc. Springfield OH 45505		**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) [1]Air Force Materiel Command Air Force Research Laboratory Human Effectiveness Directorate Warfighter Interface Division Cognitive Systems Branch Wright-Patterson AFB OH 45433-7022		**10. SPONSOR/MONITOR'S ACRONYM(S)** AFRL/RHCS
		11. SPONSOR/MONITOR'S REPORT NUMBER(S) AFRL-RH-WP-TP-2010-0002

12. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES

Human Factors and Ergonomics Society (HFES) 52nd Annual Meeting, New York, NY, September 22-26, 2008. 88th ABW/PA cleared on 28 May 2008, WPAFB-08-3431.

14. ABSTRACT

A schema is a mental structure that represents some aspect of the world. Schemata are used to organize current knowledge of the world and to provide a framework for future understanding. When a new situation or capability is encountered, a schema is brought up for comparison and then the schema is altered as necessary. Expectations are part of the mental schema and can determine the fate of a relationship, be the relationship human-human or human-other. As complex socio-technical systems become more ubiquitous and as events become more dynamic, the 'human-other' relationship is often human-automation and the automation is a decision support system. Understanding the expectations of the intended user of a capability helps the system designer and developer address those expectations which helps instill trust in the developed capability. While much has been written on trust, understanding expectations that underlie trust has been neglected. This paper will discuss three methods explored to directly elicit expectations thereby enhancing initial trust in human-decision support systems.

15. SUBJECT TERMS

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Janet E. Miller
a. REPORT UNCLASSIFIED	**b. ABSTRACT** UNCLASSIFIED	**c. THIS PAGE** UNCLASSIFIED	SAR	8	**19b. TELEPHONE NUMBER** *(include area code)*

Standard Form 298 (Rev. 8-98)
Prescribed by ANSI Std. 239.18

THIS PAGE LEFT INTENTIONALLY BLANK

Eliciting Expectations to Develop Trust in Systems

Janet E. Miller
Cheryl Batchelor
Air Force Research Laboratory
Dayton, Ohio

LeeAnn Perkins
Advanced Virtual Engine Test Cell (AVETEC), Inc
Springfield, Ohio

A schema is a mental structure that represents some aspect of the world. Schemata are used to organize current knowledge of the world and to provide a framework for future understanding. When a new situation or capability is encountered, a schema is brought up for comparison and then the schema is altered as necessary. Expectations are part of the mental schema and can determine the fate of a relationship, be the relationship human-human or human-other. As complex socio-technical systems become more ubiquitous and as events become more dynamic, the 'human-other' relationship is often human-automation and the automation is a decision support system. Understanding the expectations of the intended user of a capability helps the system designer and developer address those expectations which helps instill trust in the developed capability. While much has been written on trust, understanding expectations that underlie trust has been neglected. This paper will discuss three methods explored to directly elicit expectations thereby enhancing initial trust in human-decision support systems.

INTRODUCTION

A schema is a mental structure that represents some aspect of the world. Schemata are used to organize current knowledge of the world and to provide a framework for future understanding. When a new situation or capability is encountered, a schema is brought up for comparison and then the schema is altered as necessary. Expectations are part of the mental schema and can determine the fate of a relationship, be the relationship human-human or human-other. Expectations are important in reasoning as they form the groundwork for decision-making. As discussed in Pomranky, Dzindolet & Peterson (2001), events that occur which are against expectations are remembered over time. Therefore, understanding the expectations of the intended user of a capability helps the system designer and developer address those expectations which can help instill trust.

As complex socio-technical systems become more ubiquitous and as events become more dynamic, the 'human-other' relationship is often human-automation and the automation is a decision support system. Many factors affect the first impression by a human of a particular decision support system, including similarity to other systems previously experienced and aesthetics. Initial impressions can be finessed with glitz, state-of-the-art technology and even human-like interactions such as avatars. However, every system developer, program manager and system designer wants to ensure the customer has a long term relationship with the developed and delivered capability.

A myriad of methods exist (Potter et al., 2000) to discover requirements needed to perform tasks so that better support for these largely cognitive activities can be developed. The complexity of understanding the environment and the tasks, combined with the fact that experts performing cognitive tasks have difficulty reliably articulating about the task, makes requirements discovery difficult.

Despite having these established methods of gaining understanding about a domain, the systems engineering community struggles with the difficulties of handing the research results to designers and also of handing designs to developers so that shared understanding of the problem and possible solutions exists. A more frustrating challenge occurs when a developed system is implemented but is not enthusiastically embraced by the end-user. A solution to these hand-offs is participatory design (PD). PD is an established, diverse research and practice area and has a goal of engaging researchers, designers, developers, practitioners and end-users in the various activities leading to the successful development and implementation of systems. PD is an umbrella methodology which includes studies, theories, conferences and practices (Schon, 1983; Muller & Kuhn, 1993). Using methods which involve all of the stakeholders in the cradle to grave development of a system improves understanding of everyone's expectations and improves the trust relationship as all have had their input to the system. Having a trust relationship between all stakeholders in the development and implementation of a system is important to encourage acceptance of the resulting system.

Trust

Trust itself has been investigated and reported in many conference proceedings and journal articles (Dzindolet et al., 2003; Hill et al., 2006; Jonker et al., 2004; Lee & See, 2004). Trust has been defined in many ways such as Madsen & Gregor, 2000:

> "Trust is the extent to which a user is confident in, and willing to act on the basis of the recommendations, actions, and the decisions of a computer-based tool or decision aid."

A universal definition by Elofson on trust is "the reliance upon the characteristics of an object, or the behavior of a person in order to achieve a desired but uncertain objective" (1998). From Wikipedia, "Trust is a *prediction* of reliance on an action, based on what a party knows about the other party. Trust is a statement about what is otherwise unknown -- for example, because it is far away, cannot be verified, or is in the future." This last definition supports the concept that expectations are an integral part of trust.

Jian, Bisantz, & Drury (2000) formed a scale to measure trust in automation by a three-phase experiment comparing trust and distrust among several types of expectations. They identified twelve factors similar between humans and automated systems. Building on this theory, Lee & See (2004) defined appropriate reliance on automation and the differences between human-human and human-automation trust that are treated the same by the user. In terms of trusting automation, *over trust* exceeds the systems capabilities, *distrust* or falls short of the systems capabilities, and *calibrated trust* is the ultimate goal which matches trust with the systems capabilities.

Perceived reliability has also been shown to be vital in human aspects of sense and response logistics (Hill et al., 2006). Ezer, Fisk & Rogers (2007) investigated the effects of age and non-reliance costs on expected automation reliability and to determine if this expectation influences subsequent automation reliance. Their research did not find a significant difference in the expectations of older and younger adults.

The ability of a system or component to perform its required functions under stated conditions for a specified period of time defines reliability. Reliability is a critical attribute of performance which is important for building trust in a system or relationship. Reliability is an example of an expectation that underlies initial trust. Lewandowsky, Mundy, & Tan (2000) conducted a study experiment pertaining to trust and reliability. Participants underwent the process of pasteurizing orange juice in three stages, only two of which the user could choose to use an auxiliary aid. Results revealed when the automation failed, trust and self-confidence were decreased in the system. Once the human perceived the automation as reliable again, the trust and self-confidence in the system increased.

Predictability of automation plays an important role in trust of systems. Predictability is the matching of performance with expectations. If the user can predict what the automation should do, then the user can adequately assess when the system fails and how to perform the allocation without the automation. Experience, which establishes a pattern of predictability, is another important aspect of trust in automation. A study on user experiences with photocopiers was executed to expose the importance of experience in trust (Jonker et al., 2004). If the user started out with positive experiences with the photocopier, the level of trust was high. On the other hand, if the user began with negative experiences, the level of trust was low as the expectation was of unreliability. Thus, experience is significant for developing trust in automation. Once the user's experience develops in a positive perspective, self-confidence will also build in the automation forming strong bonds.

In understanding concepts of trust, teams have been researched especially as teams are being used more to address one-of-a-kind situations. Initial impressions of team members are made based on normal human interactions. For example, communication between team members has been recognized as being important in building relationships and reaching goals (Cooke, Gorman, & Kiekel, 2008). As with human-decision support systems, if the first impression is not fully positive, the relationship may be repaired over time. This mending of a relationship may not occur and, as is known, it takes just one bad experience to lose a user's advocacy.

In between the initial impression and the longer-term trust relationship, a hurdle exists which has not been fully explored, that of understanding expectations. As stated above, events that occur which are against expectations are long remembered. Therefore, expectations held by all of the stakeholders in the development and deployment of a system needs to be revealed and understood. This includes the expectations of the designer, the programmers, the developer, the program managers and the users of the actual system. Eliciting expectation at the beginning of each stage of development and deployment helps support the critical element of trust. Methods to enhance initial trust by defining expectations were explored and are discussed in the following sections.

METHOD

The methods described below were used as a basis for the Air Force Research Laboratory (AFRL) program, Commander's Predictive Environment (CPE). The program was funded with two main goals. The first goal, which has the purpose of strengthening AFRL's capabilities to develop human-centered capabilities, is for a strong bi-directorate partnership. The two directorates are the Human Effectiveness Directorate and the Information Directorate. The application of this partnership is to work toward the second goal that of building tools to enhance senior commanders' decision making process by supporting their ability to envision future operational environment options. The description given for this second goal was very grand and very broad in vision, making program success difficult. Therefore, the CPE Program Management Office decided to use a variety of methods to define the program by engaging Air Operations Center experts upfront in requirements identification and concept development as well as during the development of the potential automated capabilities. The program managers and the software developers were also involved in the methods described below. By using these methods, the CPE program co-managers established a better understanding of everyone's expectations for the resulting decision support systems and the team's relationship to avoid the frustration of developing capabilities that are not enthusiastically embraced by the end-user.

Method 1: Pre-Mortem

Pre-Mortem refers to discussing the end result of a program before the effort has begun. Team members are asked to envision the likely outcome of the program which makes them think about what might influence reaching that end. A Pre-Mortem was done at the initial meeting of the CPE team whose members are from two geographically separated locations. Participants also included those not in the research and development of the system but who were representative of end users. In order to identify the critical issues, the participants were asked to answer two hypothetical questions which bookend the two potential outcomes of the program. The participants were only given five minutes for each question so that they did not have time to evaluate their responses and so would be more of a subconscious response. Each participant answered in round-robin fashion until all responses were gathered. Discussion was not held until all responses were in.

Question 1: It is 6 years in the future. The Program team has just been presented with the Outstanding Scientist Award for the technical breakthroughs and outstanding customer support over the last 6 years. What was the single most important decision made in today's first technical working session, and what was the single most significant factor during the succeeding time period that led to this award?

Question 2: It is two years in the future and the Program has been cancelled for a total lack of progress in advancing its vision. What was the single worst decision made in today's technical working session, and what was the single most significant factor during the following years that led to this disaster for the program?

Method 2: Value-focused Thinking

Value-focused thinking (VFT) is a multi-attribute utility theory methodology that can help identify what is needed in an interface for a particular application and can be used to compare different potential interface solutions. Fully describing the methodology is beyond the scope of this paper so the reader is invited to read Keeney (1992). The methodology provides a means to reveal and address the multiple objectives of an interface design effort and includes eliciting from all stakeholders. The primary benefit that VFT provides is its ability to identify and convert the goals of a project or values of an organization into an objective realm. Its structure lends itself to handling multi-objective problems even if the objectives are of a subjective nature. Using VFT, high-level objectives are broken down into smaller values. In general terms, a VFT methodology uses a five-level delving by asking 'what is valued' several times to get at the basic rational, or in this case, expectations of a person or group. Once articulated, the values can be measured and put to a common scale, allowing their contribution to the overall objective to be evaluated. By assigning quantifiable measurements to the components, the multi-objective goal can be evaluated. Value focused thinking (VFT) is a proven decision analysis methodology that can be applied to a variety of multi-criteria situations.

This methodology was used in two ways. One was to reveal the expectations of the potential users of a developed system. The other was to reveal and reach agreement on the expectations of the program. In both cases, the discussions were led by an objective facilitator and the participants included all identified stakeholders including the system researchers, program manager and a group of intended users of the system. The facilitator asked members to clearly identify their expectations for the intended system and ensured good communication methods so that all had a chance to participate.

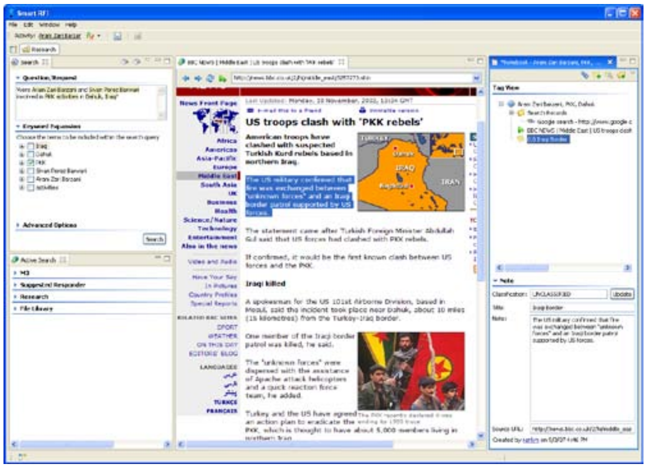

Figure 1. Prototype tool for intelligence analysts

Method 3: Vignette Framing

Vignettes, or simplified scenarios, are used to help a person get their head around a particular situation or problem. In this case, a vignette was used to elicit previously unstated expectations concerning a prototype tool to support military intelligence requests for information (Figure 1). The vignette described a military intervention scenario where the US military was tasked to establish a no-fly zone and establish and sustain air superiority under the complications of lack of near-by basing and political constraints. The intended users of the system had to envision using a tool to help them gain information concerning such as the political objective of the rebel and pro-government forces, the likely location of SA-2 batteries, and the warfighting equipment of the rebel forces. The intended users mentally went through the stages of receiving requests and envisioning how they would use such a tool to do their job while program managers and developers listened and made notes.

RESULTS

Method 1: Pre-Mortem

Although many of the responses were light and "tongue-in-cheek," they helped to identify the really important expectations of a joint R&D program. Summary of responses for explaining reason the Program was an outstanding success:

- We agreed to truly collaborate instead of just consulting with each other.
- The team streamlined the process for knowledge engineering, design, AND development.
- The research created scientific value in several technology areas.
- The team pushed the science, advanced the state of the art, and succeeded in producing something commanders can use.
- We made and followed through on a commitment to advance scientific knowledge for predictive analysis.
- We delivered something the customer loved.
- We produced a coordinated program plan that included a top-level view of all environments.

Summary of responses for why the Program was a failure:

- We failed to agree upon and implement an executable program.
- We failed to agree upon a process and framework for collaboration in the team
- We lacked clarity of vision and failed to agree upon an end product.
- We couldn't get past status quo organizational and funding issues.

Method 2: Value-focused Thinking

The fundamental objective was to identify what is valued in a software system for a complex analytical domain (Level 1). The next interaction determined that the input process of software, the processing part of software, and the output process of software (Level 2) were necessary components of the system. These two levels did not probe into expectations but helped focus the group on the future system. Level 3 elicited what were the users expectations for these Level 2 and the responses included simplicity, pleasing presentation, intuitive feel, observable engine process, simple user control, and choice of delivery. Eliciting Level 4 pushed the users to further define the responses in Level 3. Terms included in Level 4 included aesthetics, forgiveness, efficiency, flexibility, similarity to other domain software, traceability of the engine process, comprehension of the engine process and confidence.

Level 5 pushed even farther into what the user's expectations were and obtained the terms of directed input, interpretation, error alert, impact, reliable automated features, readability, attention-directing, customization, consistency, logical ordering, consistent context and readability. For a complete description, refer to McGee (2003).

Method 3: Vignette Framing

An area greatly discussed was the lack of specific requirements in the system being currently used. Gathered comments included:

- No advocacy for requirements to be filled
- Nothing with certain classifications can be entered
- Unclassified requirements not handled
- If required information is outside of the normal, defined process flow, there is no traceability
- Having a tool that can handle a variety of domain types would be excellent
- Building situation awareness quickly is needed

DISCUSSION

System developers are often stymied when systems they develop for needs they have identified never become strongly embraced and oftentimes not used at all. Participatory design methods demand that all of the stakeholders be involved during the whole process and have been shown to address this challenge. The three methods described above were used as participatory methods and revealed interesting expectations. One important insight was that the intended users of the automation wanted to have insight into how the algorithm was producing the output. The developers had expected that trust in their capability to develop the system was sufficient and had not planned on providing transparency.

The Pre-mortem allowed an atmosphere of informality and congeniality thereby allowing comments to be uninhibited. Discussion followed comments as each participant stated his fears and hopes. The questioning and delving in the VFT session at times got a bit uncomfortable as the user was made to think hard. However, working through the challenges was beneficial and the format allowed a free exchange of information. The Vignette Framing allowed the users to do a cognitive walk-through of the system with developer stakeholders present. Again, an informal, congenial atmosphere pervaded which helped the exchange of expectations. The result of using these methods was that the user community built a level of trust with the researchers and developers because the expectations were revealed and discussed which passes on to the applications being built. The intended user's expectations of the system were calibrated with the developers ability to meet those expectations thereby laying a foundation for trust in the application.

Expectations are important as they are the hidden agenda in many actions. As stated in Huron, "Expectation is an omnipresent mental process; brains are constantly anticipating the future." Expectations facilitate three purposes (Huron, 2006):

- Motivation to take action to increase the likelihood of positive outcomes: In the pre-mortem, causes to negative outcomes were identified so that changes could be proactively made.
- Preparation to react in appropriate ways: The vignette framing allowed a mental preparation to state what the tool would be expected to support.
- Representation of expected events for evaluation of the various mental representations: The VFT method encouraged all to think through events with respect to tool usage to state what they valued in these situations.

Decision support developers are at the core of trust for these systems whether the user of the system realizes that or not. When a person first uses any automation, from a television to a GPS to a blood pressure monitor, an underlying expectation is that the stakeholder developers, ranging from the programmer to the designer, understood the end users' full suite of expectations and fulfilled those expectations. Understanding those expectations and developing a system that meets those expectations builds initial trust and using a combination of methods as described above supports understanding the expectations.

Strong advocacy for the automated systems which resulted from applying these methods has been built with the intended user base. This attitude is in contrast to other attempts of implementing systems developed for the same user base. The uncovering and discussing of expectations that these methods allowed was undoubtedly part of the reason the intended community is willing to push for deployment of the applications.

DISCLAIMER

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

REFERENCES

Cooke, N. J., Gorman, J. C., & Kiekel, P. A. (2008). Communication as team-level cognitive processing. In M. Letsky, N. Warner, S. Fiore & C. Smith (Eds), *Macrocognition in Teams*. Elsevier.

Dzindolet, M., Peterson S., Pomranky R., Pierce L., & Beck H. (2003). The role of trust in automation reliance. *International Journal of Human-Computer Studies*, 58, 697-718.

Elofson, G. (1998). Developing trust with intelligent agents: An exploratory study. In *Proceedings of the First International Workshop on Trust*, 9-19.

Ezer, N., Fisk, A. D., & Rogers, W. A. (2007). Reliance on automation as a function of expectation of reliability cost of verification and age. In *Proceedings of the Human Factors and Ergonomics Society 51st Annual Meeting*, October 1-5, 2007.

Hill R., Davis J., Vohra H., Militello L., Bowman L., & Bowers, D. (2006). *Human aspects of sense and respond logistics*. Technical Report for contract F33615-99-D6. Fairborn, OH: Wright State University.

Huron, D. (2006). *Sweet anticipation: Music and the psychology of expectation.* Cambridge, MA: MIT Press.

Jian, J., Bisantz , A., & Drury, C. (2000). Foundations for an empirically determined scale of trust in automated systems. *International Journal of Cognitive Ergonomics*, 4(1), 53-71.

Jonker, C., Schalken, J., Theeuwes, J., & Treur, J. (2004). *Human experiments in trust dynamics*. Amsterdam, the Netherlands: University of Amsterdam.

Keeney, R. L. (1992). *Value-focused thinking: A path to creative decision-making*. Cambridge, MA: Harvard University Press.

Lee, J. D., & See, K. A. (2004). Trust in automation: Designing for appropriate reliance. *Human Factors,* 46(1), 50-80

Lewandowsky, S., Mundy, M., & Tan, G. (2000). The dynamics of trust: Comparing humans to automation. *Journal of Experimental Psychology: Applied*, 6(2), 104-123.

Madsen, M., & Gregor, S. (2000). Measuring human-computer trust. In *Proceedings of 11th Australasion Conference on Information Systems*, (December 6-8, 2000), Brisbane, Australia

McGee, C. M. (2003). *A value focused thinking approach to software interface in a complex analytical domain*. (MS Thesis. AFIT/GOR/ENS/03-16). Graduate School of Engineering and Management, Air Force Institute of Technology, Wright-Patterson Air Force Base, OH

Muller, M.J. & Kuhn, S. (1993). Participatory design. *Communications of the ACM*, 36(6), 24-28.

Pomranky, R. A., Dzindolet, M. T., & Peterson, S. A. (2001). Violations of expectations: A study of automation use. In *Proceedings of 6th International Conference on Command and Control Research and Technology Symposium*, Annapolis, MD.

Potter, S. S., Roth, E. M., Woods, D. D., & Elm, W. (2000). Bootstrapping multiple converging cognitive task analysis techniques for system design. In Schraagen, J.M.C., Chipman, S.F., & Shalin, V.L. (Eds.), *Cognitive Task Analysis*. Mahwah, NJ: Lawrence Erlbaum Associates.

Schon, D. A. (1983). *The reflective practitioner: How professionals think in action*. New York: Basic Books, Inc.