Engineering Complex Systems: Validating the Human Factors

George M. Samaras, Samaras & Associates, Inc., Pueblo, Colorado
george@samaras.eng.pro

Humans increase system complexity. Properly validating the human factors issues of the system reduces the degree of uncertainty in system behavior. Systems engineering employs validation to demonstrate that the proper system was constructed. Validation is based on requirements; faulty requirements result in faulty validations. Complete and correct requirements satisfice all stakeholders, inform system designers, and provide a basis for quantitative validation studies. Validation of hardware and software is well developed; validation of human factors is not! Even formulation of human factors-related system-focused requirements remains problematic. Designers continue to have difficulty integrating human factors engineering requirements into system designs. In large part, this is due to receiving user-focused, rather than system focused, requirements. A structured, systematic approach (*analysis-elaboration-synthesis*) for formulating human factors-related system-focused requirements is described. An example of a portion of a quantitative validation study for a medical device is presented. Proper formulation of human factors requirements, and their quantitative validation, is essential to enable human-centered design and effective human-systems integration.

INTRODUCTION

Introducing human actors into any system significantly increases system complexity. Un-validated, or improperly validated, systems have a high degree of uncertainty (complexity) in their behavior. Ability to predict system behavior reliably increases with increasing levels of validation. Customers [e.g., DOD, 2003] and regulators [e.g., FDA, 1996] are focusing increasingly on use and usability issues for products and processes. Proper consideration of human system integration is expected to lower the total cost of ownership of future systems through more effective, efficient, and reliable (which incorporates safer [Musa, 1999]) use. Despite this expectation, ignoring all but the most basic human factors and ergonomics (HFE) considerations remains rampant (Casey, 1993; Bogner, 2004). HFE engineers often continue to provide user-focused (not system-focused) requirements, and these requirements are not adequately operationally defined to permit quantitative validation studies.

SO, WHAT IS VALIDATION?

The engineering concept of "validation" arises from classical systems engineering (SE), which is the structured, systematic approach to the development (Research, Design, Development, Testing & Evaluation or RDDT&E), deployment, and disposal of products and processes [Hall, 1965; Nadler, 1985]. The SE space may be depicted as in Figure 1 [Samaras & Horst, 2005].

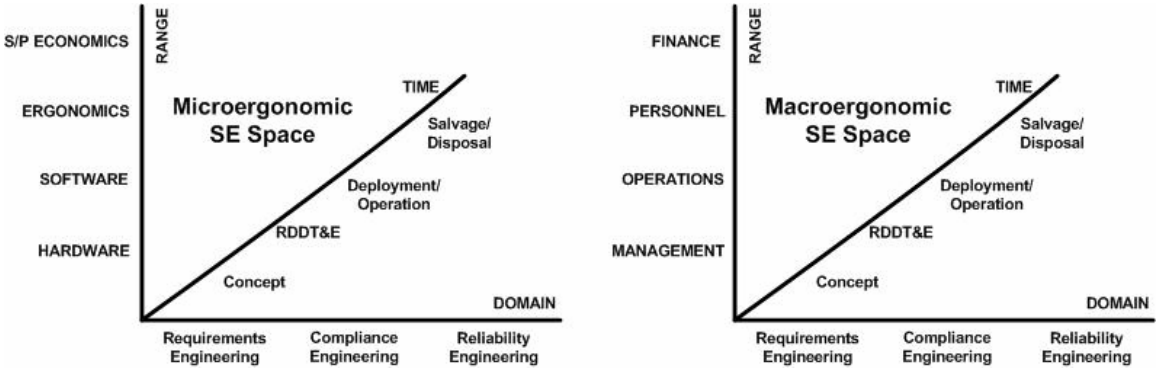

Figure 1: Micro- & Macro- Ergonomic SE Space

For microergonomics products, the disciplines range from hardware engineering to software engineering to human factors engineering to seller/purchaser economics. For macroergonomics (organizational design and management) processes, the disciplines range from management to operations to personnel to finance. The domain in both cases is the triumvirate of Requirements Engineering, Compliance Engineering, and Reliability Engineering.

From a lifecycle perspective, we can represent SE in either expanded or condensed notation (Figure 2). The condensed notation (on the right) is "less messy" and clearly delineates the verification and validation loops. Validation is about designing the correct system; verification is about designing the system correctly. The expanded notation (on the left) clearly shows the multiple iterations in the project evolution; classical SE may be highly agile with proper project management. The systems engineering process is a learning process [Eisner, 1997, pg.157]. In each iteration, some needs, wants, and desires (NWDs) are identified or discovered. Following a hazard analysis (HA), some or all of the NWDs are selected and formulated as requirements for the product or process under development. The evolving requirements (*some* → *more* → *final*) define the stakeholders' evolving understanding of the "correct design project". The HA evolves with the intermediate implementations, until the final HA corresponds to the final implementation that is successfully validated and deployed. Complete and correct requirements satisfice all stakeholders, inform designers, and provide a basis for validation measurements. Defective requirements are a principal cause of incorrect or inadequate system designs [Eisner, 1997, pg.4].

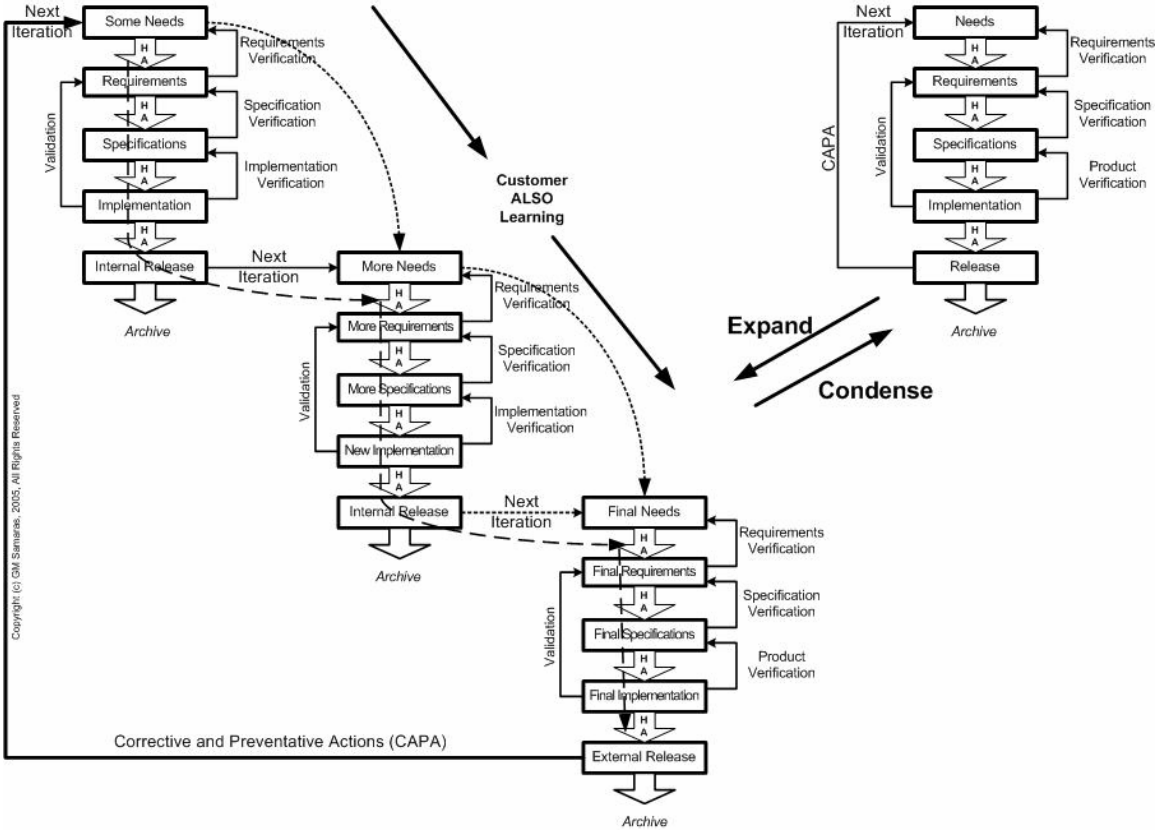

Figure 2: SE Lifecycle - Expanded vs. Condensed Notation

Validation is not essentially different from the general scientific procedures for developing and supporting theories (Cronbach & Meehl, 1955). System engineering validation (Figure 2) is based upon proper, operationally defined formulation of system-focused requirements; it consists of empirical measurements to *corroborate* compliance with these requirements. By "operationally defined", we mean you must be able to design a test for it. As the development process progresses, the set of requirements evolve with each iteration (and intermediate validations

occur) until the final set of requirements are developed (and validated) and the product or process is deployed.

It is often considered to apply only to microergonomic activities, when in fact the same SE paradigm applies to macroergonomic activities. For example, Macroergonomic Analysis and Design (MEAD) is a method of assessing work system processes for organizational design and management activities (Kleiner, 1999; Robertson, Kleiner, and O'Neill, 2002). Figure 3 shows MEAD mapped to SE (condensed notation); it expands in the same manner as shown in Figure 2. As the project evolves, the customer is learning along with the design team; this learning process is reflected in the multiple iterations. Regardless of the target system (product or process), increasing levels of validation increase the predictability and reliability of system behavior, thus reducing system complexity. The engineering of complex systems involving human actors benefits from the proper validation of HFE considerations.

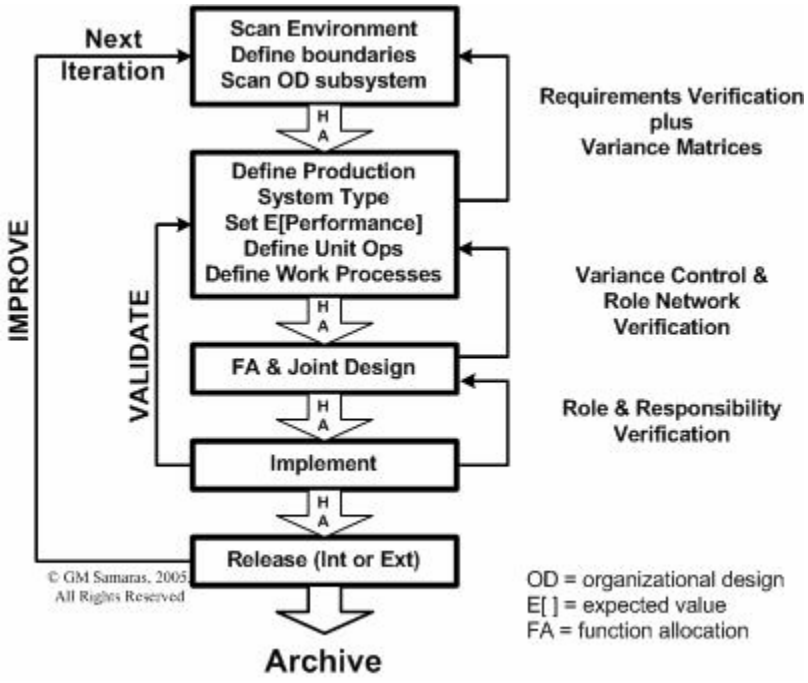

Figure 3: MEAD mapped to SE

FORMULATING SYSTEM HUMAN FACTORS

For SE to be successful, "three lines of development – the *user*, *hardware*, and *software* – have to be managed and woven into an integrated product throughout the process [Chapanis, 1996, pg. 38]." This discussion focuses on the HFE aspects of one sub-process of the SE process, *validation* and its pre-requisite – properly formulated requirements. What we mean by validation is the empirical comparison of the implementation against the properly formulated requirements. The question we are asking is "Did we build the right system?"

SE validation is based upon properly formulated requirements that operationally define an empirical study of an implementation. Consider a frequently stated "requirement" imposed upon a design team: *The system must be easy to use*! This statement is not a "requirement"; it may be an NWD, but absent operational definitions, it is NOT a requirement. An SE requirement is a natural language statement that operationally defines the validation measurement(s). Requirements Engineering is that engineering activity of discovering stakeholder NWDs, selecting those NWDs that will be translated into requirements, and formulating the requirements, so that they satisfice stakeholders, inform designers, and provide a complete and correct basis for validation. So, how do we "engineer" the NWD "*The system must be easy to use*" into a system

requirement? One approach is a process consisting of three phases: *analysis*, *elaboration*, and *synthesis*; the process repeats, in each iteration, for newly discovered NWDs.

Analysis Phase

Analysis is used in the sense of "*disclosing or working back to what is more fundamental by means of which something can be explained*" [Beaney, 2003]. The objective is to take words or expressions with complex connotations and deduce a set of elemental concepts that can be operationally defined. Users will be conceptualized in terms of structures and behaviors (Dowell & Long, 1998, pg. 132); we will consider overt and covert physical and behavioral attributes. By overt, we mean what can be directly observed (e.g., displacement); by convert we mean what cannot be directly observed (e.g., force) and can only be measured by overt resultants (e.g., second derivative of displacement). In this NWD, we have three complex terms: "system", "easy", and "use".

The term "**system**" in this particular usage does not include the user(s). From an SE perspective, the "system" is only what the designers can build. It is essential to specify the system boundaries, outside of which exists the environment over which the system designers have no control. The system could be, for example, an infusion pump for administration of intravenous fluids and medications. The boundaries form the operational definition of the "system" [Kossiakoff & Sweet, 2003]. The system, not the user(s), is the target of validation.

The term "**use**" is synonymous with "operate". It may be operationally defined as a set of specific behavioral sequences, for a specific set of conditions, which are completed within a specified time. Operation may consist of "covert physical" operation (there is no morphological component, only a biomechanical component) and/or "behavioral" operation. The term "covert behavioral" operation consists of covert observations, computations, and decisions. Their detection by "overt behavioral" operation might include gaze direction, verbal responses, and non-verbal responses (that have biomechanical & physiological characteristics).

The term "**easy**", from the user's perspective, may consist of "physically" easy and/or "behaviorally" easy. In both cases, the concept "easy" exists somewhere on the beginning of a continuum from "intuitive" through to "impossibly difficult". Before we reach the end of this continuum, more and more training and experience, will be required; however, toward the beginning ("intuitive") of the continuum, little or no training and experience will be required for acceptable "use". The term "physically" easy consists of a morphological component (e.g., size of pump front panel buttons) and a biomechanical component (e.g., syringe cassette installation force). The term "behaviorally" easy consists of an overt behavioral component (e.g., locating and pushing buttons in a certain sequence) and a covert behavioral component (e.g., deciding the button sequence).

So far, having analyzed the three original terms "system", "use", and "easy", we have no serious challenge to our current measurement capability.

Elaboration Phase

Elaboration is used in the sense of *providing additional information in intricate and painstaking detail*. Compliance Engineering activities (identifying constraints) participate in the elaboration phase. The objective is to identify clarifying and supplemental information that can be operationally defined and that constrains the requirement(s). Some examples of elaboration are:

a. Identify *who* is the user population (e.g., floor nurse, equipment repairperson) from both a morphological (e.g., age, gender, ethnicity) and experiential (e.g., Registered Nurse, Biomedical Equipment Technician) perspective. This will permit use of tabulated human perceptual, cognitive, and anthropomorphometric data and presumption of a specific range of knowledge, skills, and abilities that inform establishing training, operation, and maintenance materials. Cultural and national identifications will permit consideration of fundamental differences in conventions and expectations.
b. Identify the full range of external conditions (e.g., low light levels, mass casualties) and internal conditions (e.g., fatigue at the end of a shift, perceived time constraints) *when* the user(s) will be operating. This will permit consideration of whether the users' physical,

perceptual or cognitive capabilities may be exceeded in that environment. Make explicit seller/purchaser macro-ergonomic issues (e.g., user work scheduling, product field support, sociotechnical aspects of the development and customer organizations, etc.). Changing technology often necessitates changing organizational policies and procedures (e.g., Tate, Estes, Hagan, & Hettinger, 2005). This usually is best accomplished with a macro-ergonomic intervention (e.g., Figure 3).

c. Identify *what* are the general modes [Chapanis, 1996, pg. 10] of expected use, unexpected use, misuse, and abuse. The specific modes will identify specific behavioral sequences, for a specific set of conditions, completed within a specified time (known after each intermediate "Implementation", as shown in Figure 2, is completed and ready for validation).
d. Identify *where* the use will occur (e.g., civilian vs. military, hospital vs. field).

Synthesis Phase

Synthesis is used in the sense of *recombination of ideas into a complex whole*. The objective is to organize logically the various elements identified or discovered in the analysis and elaboration phases, so that it satisfices stakeholders and informs designers for each iteration.

Satisficing all stakeholders (e.g., customers, users, disposers, developers, producers and managers) requires understanding each group's NWDs and then prioritizing the resultant requirements, so that appropriate tradeoffs can be made in a systematic fashion. There are two general approaches: design–dependent and design independent.

Design Dependent Approach

Quality Function Deployment (QFD) is a design-dependent approach of formulating requirements. Figure 4 shows QFD in the context of the SE condensed notation. QFD employs a process of listening to the "voice of the customer" [Madu, 1999] to discover, identify and understand NWDs. These NWDs are used to develop the *quality dimension* (synonymous with SE requirements), the "whats". The "whats" are prioritized based upon their importance to each stakeholder. Putative designs (the "hows") are identified and a relationship matrix – relating the "whats" and the "hows" – is constructed. A correlation matrix is also constructed - among the "hows" – permitting identification of conflicts between putative design elements. It is important to note that QFD does not result in design-independent requirements, since putative designs participate in the selection of requirements.

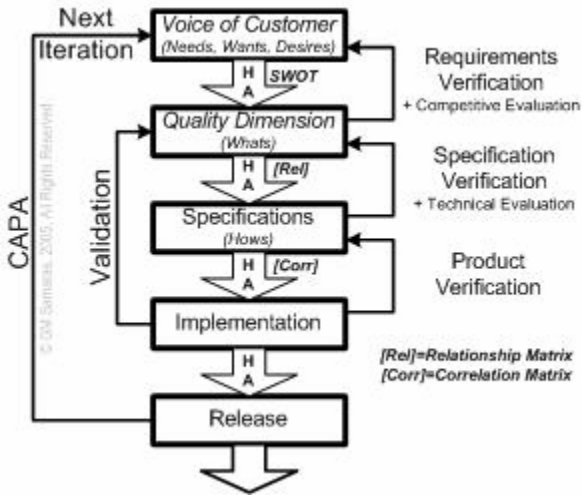

Figure 4: QFD mapped to SE

Design Independent Approach

Informing designers means that the requirements refer to the **system** [Chapanis, 1996, pg. 272], not to the user(s)! There exist at least two equivalent methods of defining design-independent requirements: (a) Use Cases that define requirements in context and (b) Requirements Specification that do not specify the use context. In both cases, the synthesis phase consists of organizing the results of the analysis and elaboration phases into a logical, understandable whole. For Use Cases, the synthesis consists of writing a "set of stories" describing the use of the system; for Requirements Specification, the synthesis consists of enumerating the system requirements in a logical, understandable document.

VALIDATING THE HFE SYSTEM REQUIREMENTS

The validation engineering activity is also not unlike the quality engineering activities of system, parameter, and tolerance design [Pyzdek, 2003, pg. 536] optimizing a process (during development) and monitoring the process (during deployment and operations). In both cases, statistically designed experiments facilitate understanding the relationship among controlled (design) factors, uncontrolled (noise) factors, and the desired output(s). Validation presupposes reliability; an important result of quantitative validation studies is Reliability Engineering data.

Let us consider a general outline of how we might approach a portion of an intermediate validation, during the development process (please refer to Figure 2), of the requirement(s) derived from the NWD "*The system shall be easy to use*". For purposes of this example, let us assume that:

a. the "system" is a single channel infusion pump for administration of intravenous fluids or medications;
b. an extended hierarchical task analysis [Chung et al, 2003] draws our attention to a particular set of pump set-up tasks;
c. naïve subjects are used (no previous experience with this particular system);
d. a maximum acceptable time (T_{max}) for correct task completion is chosen as the output threshold;
e. a 12-run Plackett-Burman [NIST/SEMATECH, 2005a] screening experiment with two replicates identifies five main factors. The design-controlled factors are display contrast, number of button pushes and the presence vs. absence of a flip card "crib sheet" with step-by-step instructions; the uncontrolled factors are syringe dimension and beginning vs. end of 12-hour shift.
f. Assuming that the controlled and uncontrolled factors are independent, an orthogonal array [NIST/SEMATECH, 2005b] design is acceptable. Assuming that the effects are linear, a 2-level study of the five factors may be used (32 runs); the dependent output is the time (T) necessary to complete the task correctly.
g. Pareto analysis [Pyzdek, 2003, pg. 17] finds that the factors and interaction effects that have most influence on the output are the display contrast and syringe dimensions.

The lack of corroboration of the implementation versus the requirements in this intermediate validation study (the result of the detected failures, $T > T_{max}$), is that two additional system requirements are introduced for the next iteration:

a. *The system liquid crystal display contrast shall have a user adjustment; the adjustment shall be large, prominent, and adjacent to the display on the front panel OR a high contrast display shall be used*;
b. *The system shall calibrate the syringe output volume, for a plunger step, prior to each infusion OR a means shall be provided to prevent the use of generic (un-calibrated) syringes.*

Note that these are human factors-related system-focused requirements, not user-focused requirements such as:

c. *The operator must be able to discriminate the displayed information at a variety of ambient light levels;*
d. *The operator must use only calibrated syringes for infusion of intravenous fluids and medications.*

Furthermore, the system-focused requirements reflect the HFE engineers' knowledge gained from the continual involvement in prior iterations (e.g., sensitivity to battery operation and power consumption concerns; sensitivity to marketing & manufacturing issues related to custom syringes).

CONCLUSION

There are no "unintended" consequences, only unanticipated consequences that are usually unwelcome! A cardinal rule of system development has to be ruthless enforcement of requirements engineering. Complete and correct requirements satisfice all stakeholders, inform designers, and provide a basis for validation measurements. This has been traditionally difficult in human-centered design. Using a structured, systematic *analysis-elaboration-synthesis* approach and maintaining a continual focus on informing system designers with system-focused requirements should improve efforts to achieve effective human-systems integration. Operationally defined, system-focused human factors requirements will inform designers and enable quantitative validation studies. Without proper validation, it is difficult to rely on a certification of completion of system development. Inclusion of HFE engineers throughout the development lifecycle will encourage incorporation of HFE knowledge and allow them to contribute throughout the complete system development process (e.g., Clark & Bonney, 2005). If HFE engineers continue to contribute only at the beginning and the end of the development process, effective, efficient, and reliable human systems integration will not be facilitated,

REFERENCES

Beaney, M. (2003). "Analysis", *The Stanford Encyclopedia of Philosophy (Summer 2003 Edition)*, Edward N. Zalta (ed.), Available at: http://plato.stanford.edu/archives/sum2003/entries/analysis/. Accessed October 19, 2005.

Bogner, M.S. (2004). Misadventures in health care: inside stories. Hillsdale, NJ: Lawrence Erlbaum.

Casey, S. (1993). Set phasers on stun and other true tales of design, technology, and human error. Santa Barbara, CA: Aegean Publishing Co.

Chapanis, A. (1996). Human factors in systems engineering. New York: Wiley.

Chung, P.H., Zhang, J., Johnson, T.R., & Patel, V.L. (2003). An extended hierarchical task analysis for error prediction in medical devices. Proceedings of AMIA 2003, 165-169.

Clark, R.D. & Bonney, S. (2005). Integration of human factors and system safety in the development of the U.S. Army tactical fire fighting truck, Proc. HFES 49th Annual Meeting, Orlando, FL., pgs. 2065-2069.

Cronbach, L.J. & Meehl, P.E. (1955). Construct validity in psychological tests. Psych. Bull, 52:281-302.

DOD I5000.2 (2003). Operation of the defense acquisition system, Enclosure 7: Human system Integration. Available at: http://akss.dau.mil/dag/DoD5002/Enc-7.asp. Accessed October 19, 2005.

Dowell, J., & Long, J. (1998). Conception of the engineering design problem. Ergonomics, 41(2): 126-139.

Eisner, H. (1997). Essentials of project and systems engineering management. New York: Wiley-Interscience.

FDA (1996). Human Factors Implications of the New GMP Rule Overall Requirements of the New Quality System Regulation. Available at: http://www.fda.gov/cdrh/humfac/hufacimp.html. Accessed October 19, 2005.

Hall, A.D. (1965). Systems engineering from an engineering viewpoint. IEEE Trans Syst Sci Cybern, SSC-1:4-8.

Kleiner, B.M. (1999). Macroergonomic analysis and design for improved safety and quality performance. International Journal of Occupational Safety and Ergonomics, 5(2), 217-245.

Kossiakoff A., & Sweet Wm.N. (2003). Systems engineering principles and practice. New Jersey: Wiley-Interscience. p.41.

Madu, C.N. (1999). House of quality in a minute. Fairfield, CT: Chi Publishers. p.9.

Musa, J. (1999). Software Reliability Engineering. New York: McGraw-Hill. p.19.

Nadler, G. (1985). Systems Methodology and Design. IEEE Trans. Syst. Man, Cybern. 15(6):685-697.

NIST/SEMATECH (2005a). Plackett-Burman Designs, NIST/SEMATECH e-Handbook of Statistical Methods, Available at: http://www.itl.nist.gov/div898/handbook/pri/section3/pri335.htm Accessed on October 25, 2005.

NIST/SEMATECH (2005b). What are Taguchi Designs? NIST/SEMATECH e-Handbook of Statistical Methods, Available at: http://www.itl.nist.gov/div898/handbook/pri/section5/pri56.htm Accessed on October 25, 2005.

Pyzdek, T. (2003). Quality Engineering Handbook. Tucson: QA Publishing.

Robertson, M.M., Kleiner, B. M., & O'Neill, M. J. (2002). Macroergonomic Methods: Assessing work systems processes, in Macroergonomics: Theory, Methods, and Applications (Hendrick, H.W. and Kleiner, B.M., Eds), Mahwah, NJ: Lawrence Erlbaum Associates

Samaras, G.M., & Horst, R.L. (2005). A systems engineering perspective on the human-centered design of health information systems, J. Biomedical Informatics, 38, 61-74.

Tate, C.C., Estes, T., Hagan, J., & Hettinger, L. (2005). Lessons learned from integrating user-centered design into a large-scale defense procurement. Proc. HFES 49th Annual Meeting, Orlando, FL. pgs. 2041-2044.

George Samaras is a consultant based in Pueblo, CO USA. He is licensed professional engineer (PE), an ASQ-certified quality engineer (CQE) and a board-certified professional ergonomist (CPE). He is an electrical engineer with a PhD (1976) in physiology from the University of Maryland and a DSc (1992) in engineering management from the George Washington University.