Persons in the Processes: Human Systems Integration in Early System Development

Nicholas Hardman[1], John Colombi[1], David Jacques[1], Raymond R. Hill[1], Janet Miller[2]
[1]Air Force Institute of Technology; 2950 Hobson Ave; WPAFB, OH 45433-7765
+1 (937) 255-6565 x7123; nhardman@afit.edu
[2]711th Human Performance Wing, Human Effectiveness Division

Abstract. The systems engineering technical processes often lack the support of methods and tools that quantitatively integrate human considerations into early system design decisions. Because of this, engineers generally rely on qualitative judgments or delay critical decisions until late in the system lifecycle. Studies have revealed that this frequently results in cost, schedule, and effectiveness consequences. We begin this paper with a study of current issues in the application of systems engineering as a whole. We then examine how the challenges of human systems integration are a pervasive factor in those issues. Finally, we propose how to improve system engineering by better addressing the identified challenges.

Introduction

Developers do not have the means to quantitatively integrate human considerations into system development. Though human systems integration (HSI) is a growing segment of systems engineering (SE) literature, studies reveal that many projects still fall short of the system effectiveness that is achievable if human components are fully integrated in the systems engineering processes (USAF HSI Office 2008; Bainbridge 2004, 958; CSE 2008; Bias and Mayhew 2005, 660; Booher 2003; Dekker 2004; Dray 1995, 17-20; Harris and Muir 2005, 963-965; Malone and Carson 2003, 37-48; Mayhew 1999, 542; GAO 2005; Norman 2007).

Systems engineers develop systems using prescribed processes. These are interacting activities which transform inputs into outputs (ISO 9000 2005). Various communities have established standards to formalize SE processes. “Standards are meant to provide an organization with a set of processes that, if done by qualified persons using appropriate tools and methods, will provide a capability to do effective and efficient engineering of systems.” (DAU 2006). Standards delineate *what* needs to be done, but they generally do not dictate *how* to do it. The International Council on Systems Engineering (INCOSE) SE Handbook expands on the processes and process groups defined by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC) in ISO 15288 (INCOSE 2007). As it states in the first chapter, this standard is intended to be supported by methods and tools developed for particular organizations (ISO 15288 2008).

This paper proposes how to form an HSI methodology that is more multidisciplinary and empirically-based. Such an approach would address the criticisms of current methods; that they are too subjective, and that they do not take full advantage of the depth of data on human capabilities and limitations (Burns et al. 2005). Our methodology is intended to be used in the context of process activity roadmaps such as those described in the INCOSE SE Handbook, IEEE 1220, or ISO 15288 (INCOSE 2007; IEEE 2008; ISO/IEC 26702 2007). We first identify those current system development processes.

The SE technical processes defined in ISO/IEC 15288, and expanded upon in the INCOSE SE Handbook, are grouped into four process groups. One of these, the Technical Processes group, involves making technical decisions to optimize the benefits and reduce the risks during system development. These processes consist of: Stakeholder Requirements Definition, Requirements Analysis, Architectural Design, Implementation, Integration, Verification, Transition, Validation, Operation, Maintenance, and Disposal (INCOSE 2007). In Figure 1 we express these process relationships graphically in the context of the SE Vee concept. The iterative flow that is depicted is to be applied at all levels of system development so that the systems engineer can define the boundary of the problem and the top-level requirements and then decompose those requirements to sufficient detail for defining feasible solutions.

In Figure 1, the left side of the SE Vee consists of the technical processes for system design. If these processes are executed with methods and tools that facilitate quantitative analysis, then the technical processes for product realization, the right side of the Vee model, can be performed quantitatively as well. Thus, this paper focuses on the technical processes of the left side of the Vee.

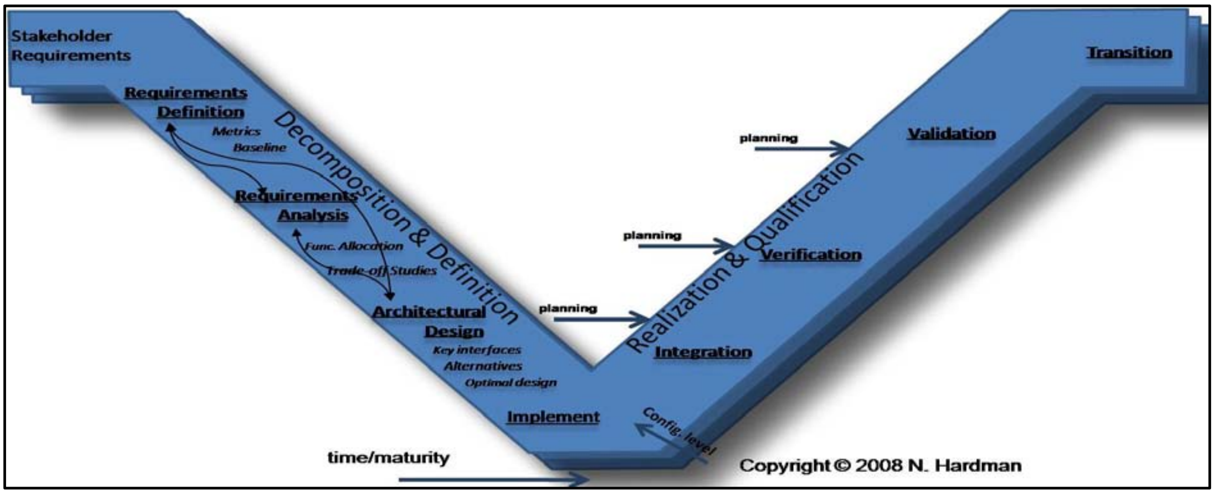

Figure1. SE Processes on the SE Vee Model

Requirements Definition. The Stakeholder Requirements Definition process is necessary in order to define the top-level requirements. During this process, the projected mission, context, and technology readiness are evaluated. Stakeholder inputs are used to define the needs and objectives; which are refined into technical requirements. The constraints on system solutions are also identified (INCOSE 2007).

Requirements Analysis. During the Requirements Analysis process, systems engineers improve the understanding of the technical requirements and their inter-relationships. In this process, top-level requirements and constraints are decomposed. System functions are then defined and allocated to system components. This creates derived technical requirements and necessary component interfaces. Ideally, the progression of these iterative steps is captured in an integrated common representation. This process enables the completion of system development in a logical manner (INCOSE 2007).

Architectural Design. The Architectural Design Solution process translates the outputs of the previous processes into feasible alternative solutions used for the final design decisions. This results in a physical design of all system components capable of performing the required functions within the identified constraints. These design decisions must be objective and traceable (INCOSE 2007).

Current Systems Engineering Issues

Systems engineering has generally been accepted as an essential part of acquisition, but many studies have found evidence of problems with the execution of the aforementioned systems engineering processes (Bias and Mayhew 2005, 660; Malone and Carson 2003, 37-48; Pew and Mavor 2007). Sage and Rouse, in their comprehensive *Handbook of Systems Engineering and Management*, conclude that SE processes are fundamentally sound, but the application of SE processes manifests many common problems. They list their “most deadly systems engineering transgressions”. Of the most critical, they list: a failure to develop and apply the appropriate methods to support the SE processes, a failure to design the system with consideration for the “cognitive style and behavioral constraints” that affect the users, and a failure to design the system for effective user interaction (Sage and Rouse 1999). Designers commit these failures when they create designer-centered systems; that is, they forget that users will not see the system with the same view. Users will only see what is presented to them at the user interfaces (Pew and Mavor 2007). The new standards reviewing committees for the ISO have also stated that there is a need for improved tools for requirements measurement and interface management. They desire to improve ISO standards as such tools are developed (Bausman 2008).

Recent studies within the United States Department of Defense (DoD) have also found inadequate systems engineering application in military acquisition programs (Bausman 2008; Saunders 2005; U.S. Air Force Studies Board 2008). A National Defense Industrial Association (NDIA) study identified areas of DoD acquisition requiring improvement. They concluded that insufficient SE is applied early in the program lifecycle, hindering initial requirements and architecture generation. They also concluded that current tools and methods are inadequate to execute the SE processes (NDIA 2003). The consequences of these deficiencies in execution have been costly and time consuming. A 2005 U.S. Government Accountability Office (GAO) report, found that major weapon systems programs averaged cost increases of 42% above original estimates and schedule slips of almost 20%. Of the identified overrun causes, the GAO analysts determined that most were the result of problems that could have been discovered early in the design process (GAO 2005). Recent assessments by the Office of the US Secretary of Defense agree with the NDIA study and state that the reason these problems are not being discovered early is that insufficient SE is applied, requirements are not well-managed, and SE tools are inadequate (CSE 2008).

A recent interview with engineers from a major U.S. company reveals that system developers recognize the need for better technical tools. The engineers were specifically emphatic when discussing human-related requirements. They stated that, without the ability to better capture and express these requirements quantitatively, they will never gain full consideration in program management tradeoff analysis; objective performance measures always dominate when contracts are at stake (Graeber and Snow 2008). Program managers continue to need actionable data early in the acquisition process. An Air Force HSI Office study has determined that timelines will continue to compress and decisions that lock in design features will increasingly be made earlier in the lifecycle (USAF HSI Office 2008).

In summary, these and other studies have investigated the causal factors of programmatic failures. They form a general consensus that the following issues exist in the application of SE technical processes:

1. sound SE *earlier* in system development
2. more SE methods and tools that provide *quantitative and actionable* data
3. more complete management of *interfaces*

HSI in SE Issues

Human components are an integral part of almost all systems, and aspects within HSI are at the heart of many of the identified systems engineering issues. The DoD recognizes this and has revised the DoD Instruction (DoDI) 5000.2-R Sec 4.3.8 to require comprehensive management strategies for HSI to assure human performance, reduce manpower, personnel, and training requirements and comply with all of the constraints for human operation (DoD 2008a). The new DoD Directive (DoDD) 5000.1 directs the program manager to have a comprehensive HSI plan for early in the program lifecycle; a plan that is reviewed at each milestone. This increased attention on HSI in the DoD policy reflects the understanding that demands on operators are increasing and changing in form. Studies by the Air Force HSI Office show that modern systems operators, surrounded by ubiquitous computer automation and augmentation, are actually experiencing greater cognitive workload (USAF HSI Office 2008). Though this seems counter-intuitive, it shows that the increase in complexity, both of mission and machine, is growing faster than technological improvements can alleviate. This is being proven out in operation in Unmanned Aerial System (UAS) deployment. The latest generation of UASs are the most automated ever, but they are also have more capabilities, serve in multiple roles, and fly in larger numbers than ever before (Lindlaw 2008). This heightens the need to properly embed all domains of HSI in the system development processes.

HSI Domains

Though there is no universally accepted delineation of HSI domains, there is much in common among the definitions found in current literature. The key components have been extracted and written in terms that enable a system engineer to clearly categorize system requirements by domain and to perform tradeoff analysis between domains. These definitions draw heavily upon those put forth by INCOSE and the Human Effectiveness Directorate of the Air Force Research Laboratory (INCOSE 2007; AIRPRINT 2005).

Manpower. The Manpower domain determines the number and type of personnel required to operate and support a system. Support includes functions such as maintenance, sustainment, and training. Many civilian organizations call this “human resources”.

DoD direction on manpower estimates for major defense acquisition programs is extensive. Program managers must coordinate with the manpower community, and the final manpower estimate is reviewed by the Under Secretary of Defense for Personnel and Readiness (DoD 1999).

Personnel. The Personnel domain determines the knowledge, skills, and abilities (KSAs) and the physical, cognitive and sensory capabilities required of the humans in the system. The personnel community defines these parameters for the system and determines how to best obtain and maintain an adequate pool of qualified people. Some military organizations call it “personal capabilities”, and in civilian organizations this domain is inter-related with human resources.

Training. The Training domain determines the necessary infrastructure and system components to provide system personnel with the requisite attributes (KSAs) for optimal system performance. This includes individual and unit training programs, training systems, and retraining schedules.

Human Factors. The Human Factors (HF) domain addresses how to incorporate human characteristics and limitations into system design for optimal usability. A primary concern for HF is the creation of effective user interfaces. Issues in this domain are often divided into the following categories:

Cognitive— e.g., response times, level of autonomy, cognitive workload limitations

Physical— e.g., ergonomic control design, anthropometric accommodation, workload limitations

Sensory— e.g., perceptual capabilities, such as sight, hearing, or tactile

Team dynamic—e.g., communication and delegation, task sharing, crew resource management

Much of U.S. industry calls this “human factors engineering (HFE)” and European and Asian organizations generically refer to it as “ergonomics”. The methods and tools of this domain are the most mature of all the HSI domains.

System Safety. The System Safety domain evaluates the characteristics and procedures of systems in order to minimize the potential for accidents. Safety studies affect system design by advocating features that eliminate hazards when possible and manage those unavoidable hazards. Such features include sub-systems for: system status, alert, backup, error recovery, and environmental risk.

Survivability. The Survivability domain evaluates the characteristics and procedures of systems that can reduce the probability of attack or fratricide, as well as minimizing system damage and injury if attacked.

Health. The Health domain evaluates the characteristics and procedures of systems that create significant risks of injury or illness to humans. Sources of health hazards include: noise, temperature, humidity, CBRNE (chemical, biological, radiological, nuclear, and explosive) substances, physical trauma, and electric shock.

Habitability. The Habitability domain evaluates the characteristics and procedures of systems that have a direct impact on personnel effectiveness by maintaining morale, comfort, and quality of life. These characteristics uniquely include: climate control, space layout, and support services.

Environment. The Environment domain evaluates the system in the medium for operation. Consideration is made to protect the environment from system manufacturing, operations, sustainment, and disposal activities. In some communities this domain is not considered part of HSI, but rather of systems engineering as a whole.

The Challenges of HSI

HSI-related system requirements are not easily quantified and the tradeoffs are often analytically complex. What is clear is the significance of such issues. DoD cost studies estimate that 40 to 60 % of the total system's lifecycle cost is determined by decisions in the HSI domains (INCOSE 2007). Air Force studies attest to the proven benefits of addressing HSI issues early in system development. Potential benefits include: reduced lifecycle cost, reduced time to fielding, shorter training cycles, improved supportability, reduced logistical footprint, and improved safety. Studies also predict that the impact of HSI issues will increase as systems and missions become more complex (USAF HSI Office, 2008). HSI must be addressed as part of disciplined system engineering. Burns, et al. state, “System tradeoff analysis is the purview of systems engineering, and thus we posit, to be successful, HSI must live within the systems engineering process.” (Burns et al. 2005). They go on to present two axioms for what HSI must be in order to address the needs of the SE community:

1. HSI must directly support the needs of acquisition.
2. The HSI community must provide analytic data consistent with the level of detail of design choices made within the systems engineering process.

The National Research Council recently reviewed the current approaches for HSI in system design. They recommended that researchers develop better ways to incorporate human-related issues into early system tradeoff analysis. Traditionally, methods and tools of the HSI community were primarily applied to the design and testing of specific components (Pew and Mavor 2007). This broadened view calls for an approach similar to prototyping or simulation.

◀ ● ▶

This calls for new methods that can be used to evaluate system alternatives and minimize programmatic risk. Systems engineers need better methods and tools to incorporate HSI into the overall technical approach. These methods and tools must help them better perform requirements elicitation, function allocation, tradeoff analysis, and design optimization. We next propose a methodology for use early and throughout the SE Technical Processes.

A Better Methodology for HSI

The SE processes can be more effective if the before mentioned issues are satisfactorily addressed. Published SE standards are essential for specifying the design processes, but they need methods to provide the means for successful system development. Methods are systematic ways of doing things, and some methods involve tools to perform specific steps of the process. A complete methodology requires more than a set of these methods and tools; it requires a unifying basis that underlies the whole approach. New system development will encounter many key decisions that will commit the design effort in a nearly irreversible course. We propose some needed elements of an improved methodology; one that will enable sound decision making early in the technical processes.

Approach HSI Empirically

Theory is essential to science. The scientific method involves creating theories that define phenomena and identify the causal relationships that explain the phenomena. In this way, theories explain what is known and predict what is unknown (Bainbridge 2004, 958). Work to produce unifying theories in systems engineering have been difficult. There is a common mantra that, “Systems engineering does not have an F=ma” (anon.); the relationship between entities are not directly connected. To date, the community primarily works with principles, heuristics, and other qualitative methods. Though a theoretical foundation of SE has proven elusive, there exist enormous volumes of empirical data regarding all domains of human systems integration. Blanchard and Fabrycky say that the improvement path for SE processes requires that they first become more quantitative and then optimized (Blanchard and Fabrycky 2006). Though this approach may not be feasible for all issues, there is still great potential in the proper engineering application of empirical data within the SE processes.

Empirical methods are not new to human factors; anthropometric studies have been used to create ergonomic design standards for decades. The data produced in the studies allow developers to make objective decisions early in the system development process. The same is possible for other aspects of human consideration in system development; albeit with much less of a direct connection. For example, in (Hardman et al., 2008) we investigated the process for menu layouts in multi-function displays. Current design efforts rely on designer intuition followed by extensive user testing. We discovered that the effectiveness of a given menu layout could be accurately predicted using existent human subjects data. This gives designers a quantitative tool for early design decisions, and it greatly reduces the necessary iterations of costly user testing. The following sub-sections give two additional propositions of how to improve SE with a more empirical approach.

Quantify Predictions. Modern perspectives in accident investigation follow the theories of Dr. J. Reason who proposed that a hazard becomes an accident through the ill-fated alignment of certain latent conditions in the systems that are in place to prevent such accidents (Reason 1990). This concept is represented in Figure 2. The data gathered by accident investigation boards on legacy systems can reveal valuable insight for the design of the next generation of systems, but only if design teams know how to use that data. The DoD has recognized the significance that HSI plays in military operations, or more specifically, in *mis*-operations. The U.S. military, and most NATO forces, have independent safety

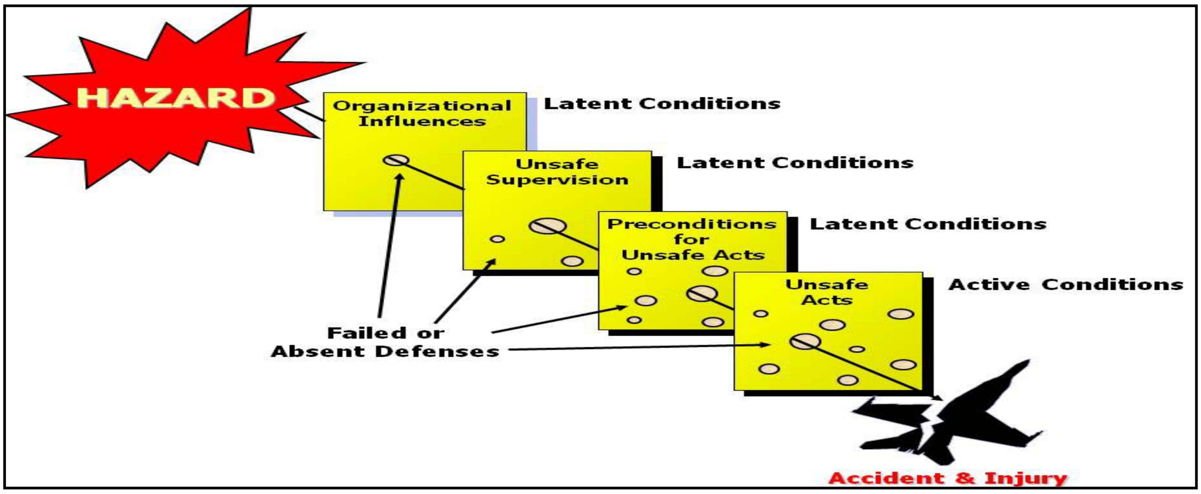

Figure 2. Accident Causation Theory (Src: AFSC, 2007)

organizations responsible for accident investigation and reporting. For the U.S. Air Force, AFPD 91-2: *Safety Programs* directs the creation of all safety programs and AFI 91-202: *The US Air Force Mishap Prevention Program* directs the mishap prevention program. The USAF Safety Investigation Board (SIB) process is guided by AFI 91-204: *Safety Investigations and Reports* (USAF 2006). These regulations give specific instruction for investigating how issues in the domains of HSI relate to specific events.

The archives of legacy system mishaps provide a large database of known problems. Studies of human error in previous mishaps can reveal how and why the human-machine interaction may fail during similar operational activities. Even very novel systems have similarities, either in structure or use, to legacy systems. Knowing what design issues have plagued past operators can guide the generation of requirements to address the identified issues. This will contribute to measures of: total system reliability, the impact of automation, and human-in-system parametrics. In a general sense, these provide quantitative justification for giving HSI domains proper attention. For example, as shown in Figure 3, aircraft mishap investigations reveal that over 65% of aircraft mishaps still involve HSI-related causal factors (AFSC 2007). With more specific analysis, design decisions in early system development can be made using quantitative data.

Figure 3. Annual Percentage of Mishaps Attributed to HSI Issues
(Src: AFSC, 2007)

Quantify User Interface Requirements. An interface is a boundary or point common to two or more entities at which necessary information flow takes place (Booher 2003). The DoD recognizes that interface management, including the UI, is a key to effective system integration (DAU 2006). Maier argues that one of the key contributions that SE should make to a program is an attention to the system's interfaces. He offers the following heuristic, "The greatest leverage in system architecting is at the interfaces. The greatest dangers are also at the interfaces." (Maier 1999, 267). Maier's heuristic aligns with evidence cited in the text, *Designing for the User Interface* regarding the interfaces of computer and machine (Shneiderman and Plaisant 2005). Dray identifies a direct connection between lifecycle costs and investment in user interfaces. She cites a company project in which an improved user interface on a large-scale internal application resulted in a 32% overall rate of return stemming from a 35% reduction in training and a 30% reduction in supervisory time (Dray 1995, 17-20). Savings such as these continue throughout the life of a system and become a significant factor in controlling total ownership cost. Given this criticality, systems engineers must have a methodology that properly focuses on UI design.

As computers become more ubiquitous, systems engineers recognize the immense significance of the interaction of operators and computers. Formal study of this has matured and expanded in perspective over the last three decades. It is now generally referred to as human-computer interaction (HCI). Our composite definition is that the field of HCI is "a field of study that seeks to improve the relations between users and computers by making computers more usable, intuitive, and accommodating of human capabilities and limitations." Many publications use HCI-related terms in an inconsistent manner. The DoD has listed standard terms and a taxonomy for these terms in the military handbook (MIL-HDBK) -1908B: *DoD Handbook on Definitions of Human Factors Terms* (DoD 1999) and MIL-STD-1472: *Human Engineering* (DoD 2003). Several professional societies have also defined their standard terminology, but none has gained universal acceptance.

Though the terminology varies by community, the concepts are similar. The central emphasis of HCI, by that or any other name, is the design of effective user interfaces (UIs); that is, the multi-modal exchanges between a human being and hardware. These interfaces facilitate interaction between human cognition and software logic. This is a critical part of HSI. Figure 4 is a conceptual depiction of human factors, HCI, and UI design. The theories and methods of the HCI community can be very useful in environments where the demand for highly effective operator performance is paramount.

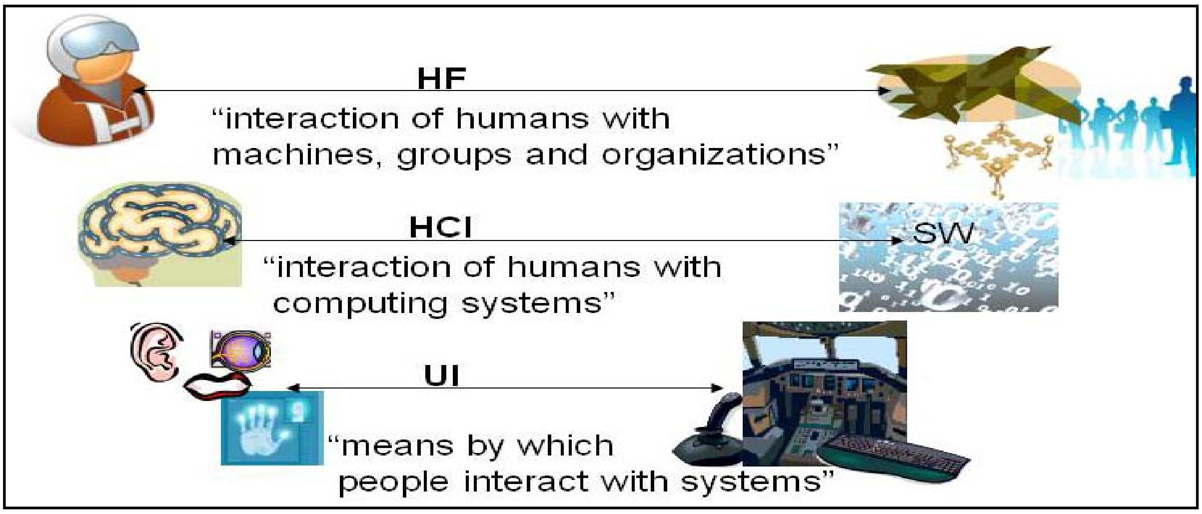

Figure 4. Human Factors Domain Components

HCI requirements are less concrete than other system requirements. HCI experts call the measure of a user interface its *usability*. Various sources expand on the concept of usability in different ways, but it is common to define the measure using the following five components (Wickens et al. 2004):

Reliability -- The frequency of errors, the prevention of catastrophic errors, and the ability to recover from errors.

Efficiency -- The level of productivity that can be achieved once learning has occurred.

Learnability -- The amount of training necessary before the user can be productive.

Memorability -- The ability for an infrequent user to maintain proficiency.

Satisfaction: -- The subjective experience of the user.

These last three components are unique to the user interface. Because we cannot yet simply program humans, interface learning time and recall ability are essential interface metrics. Standard HCI texts, such as the *Berkshire Encyclopedia of HCI*, address how to measure the five components of usability (Bainbridge 2004, 958). With the use of the proper benchmarks and surveys, these usability metrics can be standardized and quantified for use in system development. This is discussed in (Hardman et al. 2008, 19-23). While usability evaluations make UI analysis quantifiable, designers have a need to predict the potential usability of a configuration even earlier in system development. Usability analysis cannot be performed until actual systems or prototypes are developed, but empirical human data models can make projections of UI usability much earlier. This is akin to the use of wind tunnels to predict the performance of an aircraft wing even before an actual one is built. Such information could equip systems engineers to effectively address HCI early in system development.

Develop a Requirements and Constraints Paradigm

When incorporating HSI domains into SE technical processes, not all domains should be addressed the same. The human factors, manpower, personnel, and training domains can be effectively managed as design variables that interact with other system requirements. Issues within these domains are often inextricably connected, and system tradeoff analyses must include those inter-relations. For example, as described above, program managers must make early estimates regarding manpower requirements. However, the manpower estimate for a system under development is highly sensitive to decisions made in the personnel, training and human factors domains. This means the manpower estimate must concur with the analysis contained in many tangential documents throughout the system development process.

Conversely, the domains of safety, survivability, health, habitability, and environment form constraints on the system. The analysis of requirements and constraints due to human considerations is a multi-dimensional optimization problem. A useful perspective is to consider the analog of a physical volume as portrayed in Figure 5. Within the context of the system's intended mission, the constraints limit the size and shape of the trade space. The necessary performance of the system creates an analogous volume with manpower, personnel, and training dimensions. Systems engineers must meet the mission requirements, the volume, in such a way that it fits within the contextual constraints, the space. Human factors issues have the effect of either increasing the volume or filling it. Poor human consideration can increase demands in the manpower, personnel, and training dimensions; alternatively, improved human factors can be effective force multipliers by reducing operator workload. This paradigm defines an approach that can make tradeoff analysis more complete, context-aware, and objective.

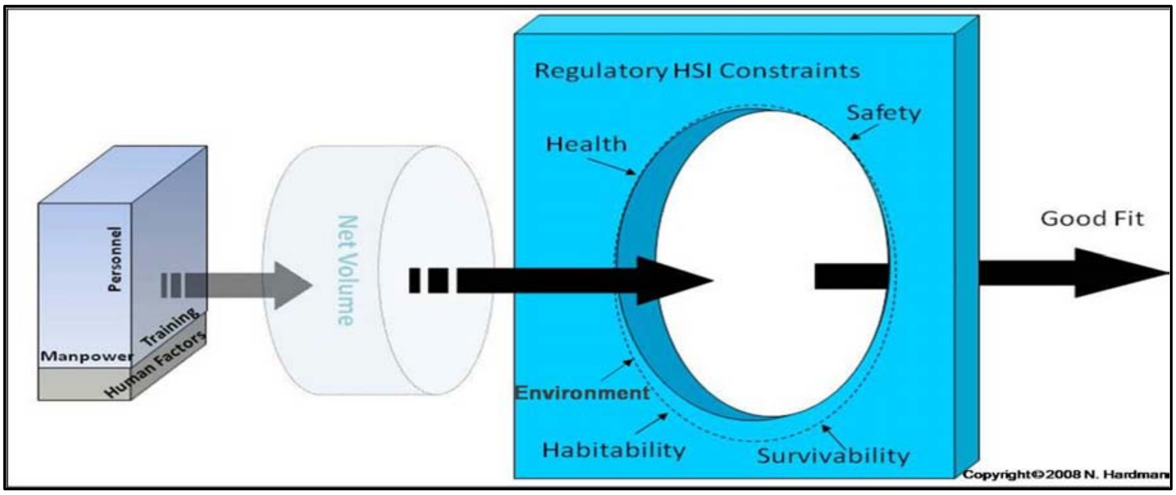

Figure 5. HSI Requirements and Constraints

Integrate with SE Technical Process

The purpose for an improved HSI methodology is to provide actionable data to make timely design decisions. To do that, it must be embedded within the SE technical processes for design.

Requirements Definition. As mentioned, HCI practitioners study how to best capture system interactions. The HCI community can use these methods to predict areas for design emphasis in order to avoid what Hoffman and Elm call the "pitfalls of designer-centered designs" (Hoffman and Elm 2006). Even more lucid insights are possible if prototypical problems have been cataloged. The Requirements Definition process is the proper time to perform an empirical study of legacy system mishaps involving human error as a causal factor. This is similar to the failure analysis already used to provide feedback to the acquisition community regarding structural and propulsion aircraft components. This may also identify additional human subjects research that may be needed to prepare for the following processes.

Requirements Analysis. During the Requirements Analysis process, a significant amount of effort is required for decomposition and function allocation. A function is a logical unit of behavior of a system (Blanchard and Fabrycky 2006). Function allocation is "a procedure for assigning each system function, action, and decision to hardware, software, operators, maintainers, or some combination of them" (NASA 2007). This is normally a process that involves more art than science. Humans, as components of the system, must be a part of function allocation decisions.

When it concerns humans and computers, function allocation involves the study of automation. Many function allocation decisions are made at the micro level when components are selected. This often leaves the designer with a limited number of highly constrained explicit allocation decisions. Function allocation influences all domains of HSI because they affect the number and location of operators. Therefore, function allocation decisions precede other SE tasks such as: task analysis, operator workload analysis, and UI design. Function allocation must be done in a manner that optimizes performance and safety. If one uses empirical data for function allocation decisions, those decisions will be more traceable and will enable easier adaptation of new missions or technology. Previous efforts in this area were highly influenced by the famous 1951 Fitt's List that itemized what computers did better versus what humans did better (Sheridan 2000, 203-216). The problem with this approach is twofold: First, a division along such lines will quickly be made obsolete by the changing landscape of technology. More significantly, this approach takes an incomplete view of the problem. At its core, the challenge is to understand the required tasks and how to support the humans that must do them. This requires more information regarding the given context and the desired outcome. We can draw from empirical data from basic research in physiology and psychology to clarify these decisions. If done in a context-aware approach, i.e., one that accounts for mission and medium factors, this method will yield quantitative answers for automation decisions. For example, designers of unmanned aerial systems (UASs) desire to examine the automation of detect, see and avoid (DSA) tasks. They desire quantitative measures of the performance differences between humans and automated systems. In (Hardman 2005) this was analyzed using existing human subjects and sensor data.

Architectural Design. The Architectural Design process includes determining the feasibility of alternatives and ultimately choosing the final design solution. This can be a complicated sequence that must be done early and throughout system development. Usability analysis and model-based predictions of usability can support alternative evaluation.

One area where this approach holds promise is in the design of input devices. Many systems require dynamic inputs from humans. This includes manual tracking tasks such as slewing a target for object selection, panning and zooming, and the teleoperation of robotic vehicles. Systems engineers must assure that the interface between the human operators and the machine is not a performance limiting factor. These interactions have the following characteristics: the operator knows the goal, the system possesses the means to achieve the goal, there is a speed and accuracy tradeoff of communicating the goal to the system, and the operator receives feedback regarding performance. The interface must account for the parameters of both the machine and the human. Control systems texts offer very complete procedures for determining the response parameters of machines, but human-inclusive determinations are less defined.

Another area that needs to be addressed is user displays layout design. Display design begins with specifications listing all necessary information and configuration requirements. The designer must find the best layout to satisfy these specifications. The layout should follow a context-aware design paradigm in that the informatic relationships are based on what is needed for a given operational activity thread (Dix et al. 2004). For example, for aviation applications, experimental psychologists have correlated necessary information by phase of flight (Schvaneveldt, Beringer, and Lamonica 2001, 253-280). This information can not only improve early design suitability evaluation, it can greatly reduce the effort required for alternative analysis. This was the impetuous for the approach, as detailed earlier, to predict display layout effectiveness (Hardman et al. 2008).

Summary

We have summarized the current SE challenges and examined how HSI is a pervasive factor in these challenges. We then proposed how to improve system engineering through a methodology that would address the identified challenges. Throughout the discussion, we referenced specific work being done using such a methodology. This approach can bridge empirical research on human capabilities and the challenges of SE in early system development.

BIOGRAPHY

Nicholas Hardman is a doctoral student in systems engineering at the Air Force Institute of Technology. He has over ten years of experience in various positions including F-22 special projects test manager, F-15 avionics flight test engineer, data links test flight commander, and lead engineer for allied unmanned aerial vehicle development. Nick has a BS and MS in Electrical Engineering, and an MS in Aviation Systems.

John Colombi, Ph.D. is an Assistant Professor of Systems Engineering at the Air Force Institute of Technology. He teaches graduate courses in software, systems and enterprise architecture and leads sponsored research in Human Systems Integration. Before joining the faculty, Dr. Colombi led command and control systems integration activities including systems engineering for the E-3 aircraft at Hanscom AFB. He has served at the National Security Agency developing biometrics and information security and ran communications networking research at Air Force Research Laboratory.

David Jacques, Ph.D. is an Assistant Professor of Aerospace Engineering on the faculty of the Air Force Institute of Technology. He is the Curriculum Chair of the Graduate Systems Engineering Program, and his research interests are in the area of concept definition and evaluation, architecture modeling, and optimal system design. Dr. Jacques led the activation of the Air Force Center for Systems Engineering at the Air Force Institute of Technology.

Raymond R. Hill, Ph.D. is a Professor of Operations Research. His research interests are in the areas of applied statistics, design of experiments, agent based modeling, and heuristic search methods. He is an associate editor for Military Operations Research, Journal of Defense Modeling and Simulation, Journal of Simulation, Information Age Warfare Quarterly, and the International Journal of Mathematics in Operational Research. He is currently a proceedings co-editor for the 2008 and the 2009 Winter Simulation Conference.

Janet Miller, Ph.D. is a Senior Electronics Engineer at the 711th Human Performance Wing, Human Effectiveness Directorate. In her current assignment, she manages programs and research in areas of human-system integration, especially with regard to forecasting in an operational environment. Prior assignments include being an information systems engineer at National Air and Space Intelligence Center, Air Force Material Command and Pacific Air Forces.

References

AFSC. 2007. *U.S. Air Force aviation safety year in review FY2006.* Kirkland AFB, NM: Air Force Safety Center.

AIRPRINT. Building blocks of the airmen performance integration program. in 311th Human Systems Wing [database online]. Brooks City-Base, TX, 2005 [cited 10 Aug 07]. Available from http://www.brooks.af.mil/.

Bainbridge, William Sims, ed. 2004. *Berkshire encyclopedia of human-computer interaction*. Great Barrington, Mass.: Berkshire Pub. Group.

Bausman, Karen. 2008. *Systems engineering process improvement.* Wright-Patterson AFB, OH: The Center for Systems Engineering, .

Bias, Randolph G., and Deborah J. Mayhew, eds. 2005. *Cost-justifying usability: An update for an internet age*. Boston: Morgan Kaufman.

Blanchard, Benjamin S., and Wolter J. Fabrycky. 2006. *Systems engineering and analysis*. 4th ed. New Jersey: Pearson Prentice Hall.

Booher, Harold R., ed. 2003. *Handbook of human systems integration*. Systems Engineering and Management., ed. Andrew Sage. 1st ed. Hoboken, NJ: John Wiley & Sons, Inc.

Burns, John, Jerry Gordon, Matt Wilson, Milt Stretton, and Dan Bowdler. 2005. A framework for applying HSI tools in systems acquisition, paper no. 2281. Interservice/Industry Training, Simularion, and Education Conference, Orlando, FL.

CSE, The Center for Systems Engineering. 2008. *The current state of systems engineering. Acquistion Professional Training Course.* Wright-Patterson AFB, OH:, .

DAU, Defense acquisition guidebook. Ft Belvior, VA. 2006 [cited 20 Jan 09]. Available from http://akss.dau.mil/dag/welcome.asp.

Dekker, Sidney. 2004. *Ten questions about human error*. Burlington, VT: CRC Press.

Dix, A., J. Finlay, G. Abowd, and R. Beale. 2004. *Human-computer interaction*. 3rd ed. Essex: Pearson Education Limited.

DoD. 2008a. *DoD directive 5000.1: The defense acquisiton system.* Washington, D.C.: US Dept. of Defense.

———. 2008b. *DoD instruction 5000.2: Operation of the defense acquisition system.* Washington, D.C.: US Dept. of Defense, .

———. 2003. *MIL-STD-1472F(1): Human engineering.* 5 Dec 2003: US Dept. of Defense, MIL-STD-1472F(1).

———. 1999. *MIL-HDBK-1908B: Definitions of human factors terms.* Washington, D.C.: US Dept. of Defense, MIL-HDBK-1908B.

———. 1998. *MIL-HDBK-759C: Human engineering design guidelines*. Washington, DC: US Dept. of Defense.

Dray, Susan. 1995. The importance of designing usable systems. *Interactions.*

GAO. 2005. *GAO-05-301: Assessments of selected major weapon programs.* Washington, D.C.: US Government Accountability Office, GAO-05-301.

Graeber, D., and Michael Snow. 2008. Interview with boeing phantomworks engineers on HSI requirements. 5 may, Seattle, WA.

Hardman, Nicholas S., J. Colombi, D. Jacques, and R. Hill. 2008. Improved user interface design through graph-theoretic modeling. *Proceedings of IEEE International Conference on Distributed Human-Machine Systems*.

———. 2008. What systems engineers need to know about human-computer interaction. *INCOSE Insight* 11, (2): 19-23.

Hardman, Nicholas S. 2005. Approaches for Autonomous Vehicles in Civil Airspace, Master's Thesis, University of Tennessee.

Harris, Don, and Helen C. Muir, eds. 2005. *Contemporary issues in human factors and aviation safety*. Vol. 50. Burlington, VT: Aldershot.

Hoffman, R., and W. Elm. 2006. HCC implications for the procurement process. *Intelligent Systems* 21.

IEEE. 2008. *Systems and software engineering - system life cycle processes.* Institute for Electrical and Electronics Engineers, IEEE STD 15288-2008.

INCOSE. 2007. *Systems engineering handbook*, ed. Cecilia Haskins. Ver. 3.1 ed.International Council on Systems Engineering.

ISO 15288. 2008. *A system and software engineering standard.* International Organization for Standardization, ISO 15288:2008.

ISO 9000. 2005. *Quality management systems--fundamentals and vocabulary.* International Organization for Standardization, 9000:2005.

ISO/IEC 26702. 2007. *Systems engineering - application and management of the systems engineering process (also, IEEE std 1220-2005)*. ISO and IEEE.

Lindlaw, Scott. 2008. Drones' big enemy is human mistakes. *Arizona Daily Star*, 26 August, 2008, 2008.

Maier, Mark W. 1999. *Architecting principles for system-of-systems.* New York: John Wiley & Sons, Inc.

Malone, T. B., and F. Carson. 2003. HSI top down requirements analysis. *Naval Engineers Journal*. 115.

Mayhew, Deborah J. 1999. *The usability engineering lifecycle: A practitioner's handbook for user interface design*. The morgan kaufmann series in interactive technologies. San Francisco, CA: Morgan Kaufmann Publishers.

Narkevicius, Jennifer M., John Winters, and Nicholas S. Hardman. 2008. Talking the talk: Cross-discipline terminology challenges. *INCOSE Insight* 11, (2): 25-7.

NASA. 2007. *NASA/SP-2007-6105: Systems engineering handbook*. REv. 1 ed. Washington, D.C.: National Aeronautics and Space Administration.

NDIA. 2003. *Top five systems engineering issues in defense industry.* Arlington, VA: National Defense Industrial Association, Systems Engineering Division, .

Norman, Don A. 2007. *The design of future things*. Jackson, TN: Basic Books.

Pew, Richard W., and Anne S. Mavor, eds. 2007. *Human-system integration in the system development process*. Washington, D.C.: National Research Council.

Reason, James. 1990. *Human error*. 1st ed. New York: Cambridge University Press.

Sage, Andrew, and William B. Rouse, eds. 1999. *Handbook of systems engineering and management*. New York: John Wiley & Sons, Inc.

Saunders, Thomas. 2005. *System-of-systems engineering for air force capability development.* USAF Scientific Advisory Board, SAB-TR-05-04.

Schvaneveldt, Roger W., Dennis B. Beringer, and John A. Lamonica. 2001. Priority and organization of information accessed by pilots in various phases of flight. 11, (3) (International Journal of Aviation Psychology): 253-80.

Sheridan, T. B. 2000. Function allocation: Algorithm, alchemy or apostasy? *International Journal of Human-Computer Studies,* 52, (2) (2): 203-16.

Shneiderman, Ben, and C. Plaisant. 2005. *Designing the user interface*. 4th ed. Boston, MA: Addison-Wesley.

U.S. Air Force Studies Board. 2008. *Pre-milestone A and early-phase systems engineering: A retrospective review and benefits for future air force systems acquisition.* Washington, D.C.: The National Academies Press, .

USAF. 2006. *AFI 91-204: Safety investigations and reports.* Washington, D.C.: Office of the Sec. USAF.

USAF HSI Office. 2008. *Briefing to the secretary of the air force.* Arlington, VA: U.S. Air Force HSI Office, .

Wickens, Christopher D., John Lee, Yili Liu, and Sallie Gordon-Becker. 2004. *An introduction to human factors engineering*. 2nd ed. Upper Saddle River, NJ: Prentice Hall.