Available online at www.sciencedirect.com
SCIENCE DIRECT®
Journal of Biomedical Informatics 38 (2005) 61–74
Journal of Biomedical Informatics
www.elsevier.com/locate/yjbin

A systems engineering perspective on the human-centered design of health information systems

George M. Samaras[a,*], Richard L. Horst[b]

[a] Samaras and Associates, Inc., 7755 Soda Creek Road, Pueblo, CO, USA
[b] UserWorks, Inc., Silver Spring, MD, USA

Received 5 November 2004
Available online 8 December 2004

Abstract

The discipline of systems engineering, over the past five decades, has used a structured systematic approach to managing the "cradle to grave" development of products and processes. While elements of this approach are typically used to guide the development of information systems that instantiate a significant user interface, it appears to be rare for the entire process to be implemented. In fact, a number of authors have put forth development lifecycle models that are subsets of the classical systems engineering method, but fail to include steps such as incremental hazard analysis and post-deployment corrective and preventative actions. In that most health information systems have safety implications, we argue that the design and development of such systems would benefit by implementing this systems engineering approach in full. Particularly with regard to bringing a human-centered perspective to the formulation of system requirements and the configuration of effective user interfaces, this classical systems engineering method provides an excellent framework for incorporating human factors (ergonomics) knowledge and integrating ergonomists in the interdisciplinary development of health information systems.
© 2004 Elsevier Inc. All rights reserved.

Keywords: Systems engineering; Human factors engineering; Methodology; Health information systems; Microergonomics; Macroergonomics; Validation; Verification

1. Introduction

Health information systems are rapidly increasing in variety, size, complexity, and sophistication. Depending on one's definitions, health information systems can range from those running on a standalone platform, e.g., a medical device, to those involving world-wide networks, distributed databases, and enterprise-wide interoperability. Regardless, some common elements among this vast range of systems are that they have human safety implications and they have interfaces with human users. The "users" of a given interface can be patients, caregivers, or system operators, administrators, or developers. Of course, most health information systems have multiple such interfaces. Thus, there is much to be said for a human-centered approach to the conceptualization, design, and development of such systems.

Human-centered research, design, development, testing, and evaluation are the core activities of the field of human factors (or ergonomics) engineering—whose mandate is to design products and processes for human use. Individuals who are not trained in ergonomics cannot be expected to anticipate all possible uses, misuses or abuses of their products or processes [1]. Organizations that do not insist on human factors engineering knowledge having "an equal seat at the table" cannot reasonably expect to avoid potentially catastrophic, unanticipated consequences in their products and processes. The fundamental architecture for professional competence in ergonomics is defined by the Board of Certification of Professional Ergonomists [2], which is

* Corresponding author. Fax: +1 719 485 3819.
E-mail address: george@samaras-assoc.com (G.M. Samaras).

1532-0464/$ - see front matter © 2004 Elsevier Inc. All rights reserved.
doi:10.1016/j.jbi.2004.11.013

endorsed by the International Ergonomics Association. In the case of health information systems, where the role of the various human stakeholders is crucial, it is hardly surprising that the implementation of sophisticated technologies often fail due to the lack of structured, systematic consideration of human issues. While such failures are sometimes attributed to "human error," there is persuasive evidence [3–5] that the fault more often lies with inadequate system design or shortfalls in the organizational structure within which these systems are utilized.

Such failures and inefficiencies can be avoided by the thorough implementation of the methods, and over a half a century of expertise, of the system engineering discipline. While elements of this approach are typically used to guide the development of information systems that instantiate a significant user interface, it appears to be rare for the entire process to be implemented.

In this regard, it is important to note that the Food and Drug Administration (FDA) details the application of a systems engineering method (that we describe here as "SE") for manufacturers of finished medical devices. Specific guidance [6] clearly indicates the human factors implications of the regulation. As health information systems begin to have more profound effects on individual patient care [4,7], the FDA may also begin to consider them as medical devices (for example, blood banking software systems are now considered medical devices and subject to the FDA regulation).

In this article, we describe the classical SE method, emphasizing that it provides a framework for incorporating ergonomics knowledge in all phases of the interdisciplinary development process and integrating the role of ergonomists into the development team. We compare and contrast the classical systems engineering method to more recently published development lifecycle methods, pointing out that the latter represent incomplete subsets of the former. We cite two accidents involving "user error" with health information systems (radiological therapy systems where the errors resulted in overexposures that were fatal to the patients) which would have likely been avoided by a conscientious application of hazard analyses. We discuss practical matters that arise in the application of an SE approach and identify tools for implementing the various elements of the SE method. Finally, we discuss some of the macroergonomic issues involved in organizational change, so that ergonomists may be involved, from cradle to grave, in the development and deployment of products and processes.

2. Historical perspective

The term *systems engineering* dates back to the Bell Telephone Laboratories in the 1940s [8]. One of the earliest descriptions of the methodological framework for systems engineering is a paper by Hall [9]. As Sage [10] indicated "It is especially interesting to note that his paper [i.e., Hall's paper], despite its date, appears to have suffered extraordinarily little from the passage of time." The USAF issued Mil-Std 499A (now obsolete) in 1974 [11]; it describes the systems engineering process and its iterative nature [12]. Nadler [13] has analyzed the theoretical and philosophical issues surrounding systems methodology and design. System design for human interaction, emphasizing system management and methodological issues, has been an important issue in systems engineering [14]. Chapanis [15] states that his major thesis is "for a system to be successful, three lines of development—the user, hardware, and software—have to be managed and woven into an integrated product throughout" the systems engineering process. Buede [16] discusses the various equivalent development models (waterfall, spiral, Vee, and rapid prototyping), pointing out that Forsberg and Mooz have shown that "the spiral activities can be mapped onto the Vee model without swapping any activities in time." The FDA has detailed this systems engineering approach in its updated Good Manufacturing Practices regulation [17]. Blanchard [18] emphasizes the cost impact of not using a rigorous, structured, systematic approach to system development, and the iterative nature of safety engineering with its numerous interfaces to the system engineering process [19]. This is also emphasized by Kossiakoff and Sweet [20].

3. Theory

The application of ergonomics *should not* operate independently of product or process development and *should not* be viewed as standing alone [21]. It is best considered within a rigorously applied, structured, systematic development framework well-known to the systems engineering discipline. It is this framework that permits taking maximal advantage of ergonomics knowledge and expertise throughout the product or process lifecycle. It is the incorporation of ergonomics knowledge in this process, rather than the perceived stature of any particular ergonomics professional, that should engender trust in the endeavor.

4. What is SE?

SE is a structured, systematic approach to system risk reduction over the full lifetime of the system (*from cradle to grave*). It is of particular importance in new product development of complex systems. Your ability to predict system behavior reliably increases with increasing levels of validation. Un-validated systems have a high degree of uncertainty (complexity) in their behavior; *validation thus decreases the complexity of system behavior*.

SE is a proactive hazard mitigation process, maximizing the likelihood of reducing errors and time to market. It is a structured, *risk*-based, iterative approach to the research, design, development, test and evaluation, deployment, and salvage/disposal of products and processes. It is a formal process that emphasizes transparency and clarity of known objectives and constraints.

4.1. The SE space

The SE domain is the triumvirate of requirements engineering, compliance engineering, and reliability engineering. The SE range includes activities from the disciplines of hardware engineering, software engineering, ergonomics, and seller/purchaser economics; these reflect the range of activities involved in development of products and processes. The time line begins with conceptualization and ends with salvage and disposal ("lust to dust"). This space within which SE takes place is depicted in Fig. 1. All SE activities can be characterized by their placement in this space, and conversely, there is some SE activity that is pertinent to all points in this space.

As a *lifecycle* process (see Fig. 2), it begins with the initial conceptualization of the system, it is continually applied throughout the research, design, development, testing and evaluation (RDDT & E) phase, in the operational phase (with periodic re-validations), and finally, when the system is obsolete, in the salvage and disposal phase. The *feedback* loops of this lifecycle model (Fig. 2) consist of validation testing (implementation vs. requirements), verification testing (of requirements, specifications, and implementation), incremental hazard analyses (HA), and post-deployment corrective and preventative actions (CAPA). The *feedforward* loop consists of needs assessment, translation of needs to quantifiable requirements, translation of requirements to quantitative engineering specifications, translation of specifications to a product/process implementation, and the deployment of the product or process.

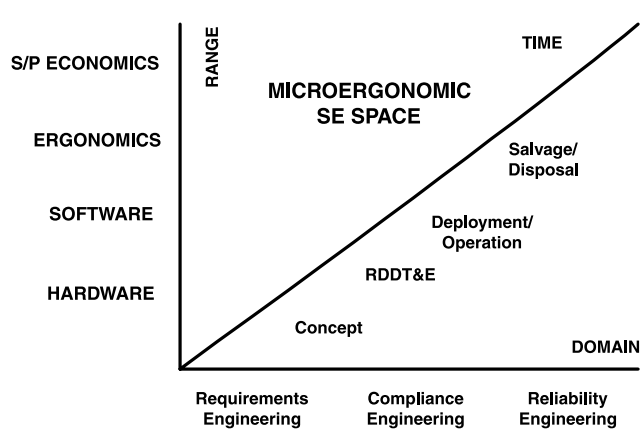

Fig. 1. Microergonomic SE space.

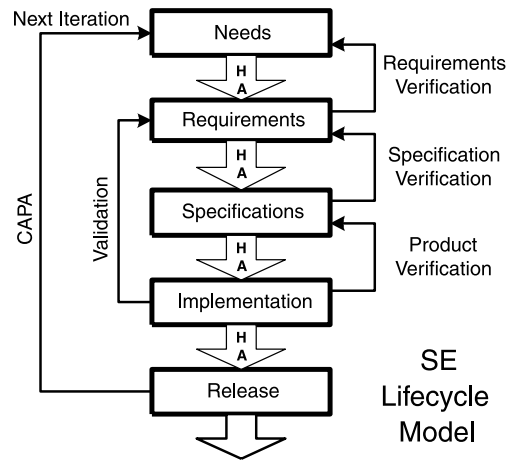

Fig. 2. The SE lifecycle model.

Ergonomic considerations participate in a manner similar to hardware, software, and economic considerations in the development of requirements, in compliance with appropriate regulations and standards, and in the engineering of system reliability.

4.2. Requirements engineering

From a product development process perspective, one can obtain a more detailed view of requirements engineering. The first step in the iterative process is identification of the needs of the system users—which presupposes that you have correctly identified the universe of user populations (manufacturers, assemblers, operators, clinicians, patients, maintainers, disposers, etc.) as shown in the Venn diagram of Fig. 3.

User needs assessment is a complex activity that often has been implemented by marketing personnel with ad hoc engineering support; in fact, it is a central area of expertise and practice in ergonomics. Some examples of needs assessment techniques include interviews, ques-

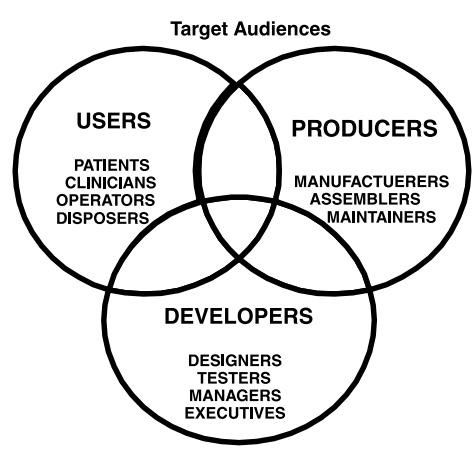

Fig. 3. The user universe—the needs of typical target audiences overlap.

tionnaires, and ethno-methodological studies, brain storming, problem-domain storyboarding, prototyping, literature reviews, and ergonomics laboratory research, as well as evolutionary (rapid and iterative) development techniques. Both from a good business practices perspective and from a FDA regulatory perspective, they must be implemented in a statistically valid manner, so that the results truly represent the populations under study.

Once the user needs have been determined, the next task is to translate the subset of needs, that will be met, into requirements of the health information system. This activity also requires the knowledge and skills of ergonomics. Requirements are the foundation of the validation process and a crucial source of the engineering design specifications (Fig. 4). When dealing with health information systems, particularly those in which proprietary software or database content run on generic hardware, the requirements and specifications may encompass such issues as response time, storage capacity, load balancing, data backup and disaster recovery, system availability, and ease of use. It is helpful to treat user interface characteristics in the same manner as these system performance variables, setting usability objectives for the system in measurable terms, typically couched in terms of effectiveness, efficiency, and user satisfaction as identified in ISO 13407:1999 [22].

Defective requirements are the principal cause of incorrect or inadequate system designs and failed validations. Common flaws include not selecting the proper target audiences and assuming you already know the user needs. Properly formulated requirements are natural language statements (e.g., English) that are understandable by the user populations, by the design team, and by seller and purchaser management. Properly formulated requirements must be traceable to specific user needs, must be clear, complete, and internally consistent, and must be verifiable (you must be able to design a test for it). In order for a requirement to be quantifiable and testable—and thus verifiable—it is imperative that there exist operational definitions of the critical elements incorporated within each requirement. Absent operational definitions, there can be no measurements and no verification.

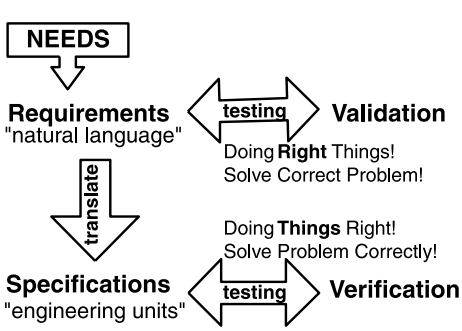

Fig. 4. Verification versus validation

Proper requirement formulation is an inter-disciplinary engineering activity that necessarily includes ergonomics expertise to represent properly the discovered needs of the various user populations. A central activity of ergonomics is translating user needs into requirements (and then requirements into engineering specifications). If this is reminiscent of "concurrent engineering" discussions, it is because it is the same discussion [23,24]. Just as electronics engineers must make sure the mechanical engineers leave enough room for their printed circuit boards, and the software engineers make sure the electronics engineers put enough memory in the circuitry for their code, and the manufacturing engineers make sure standard parts are not replaced (without good justification) with custom parts, so ergonomists make sure the design team meets the users' actual needs. And, when difficult engineering trade-offs are encountered, the ergonomists on the design team must ensure that the user's needs are properly considered—because if they are not met, either the product will fail, will produce unreliable results, will encourage new competition in the market, or worse, will harm patients!

Once the requirements are properly established and verified against the user needs, the next task is to translate these natural language statements into engineering design specifications. Engineering design specifications are the true basis for the product design and are quantitative product attributes with associated units and tolerances. Once again, the ergonomist can play a crucial role on the design team, directly impacting the work of the rest of the team and the final design of the product:

1. From a *hardware ergonomics* perspective, the ergonomist not only has access to tabulated human cognitive and perceptual data, and as appropriate, anthropometric data, which can dictate physical specifications, but the ergonomist is trained to properly use these data in the realization of engineering designs.
2. From a *software ergonomics* perspective, the ergonomist is trained to participate in the design of user interfaces, to conduct task analyses on the proposed logical operation of the product, and to participate in the design of training, operation, and maintenance materials.
3. From an *environmental ergonomics* perspective, the ergonomist can assist the design team in assessing how known workspace environmental modalities can impact the use and reliability of the proposed design (e.g., effects of temperature, humidity, lighting, ambient noise, and air quality on user fatigue, perceptual, and cognitive abilities).
4. From a *macro-ergonomics* perspective, some ergonomists can assist the organization in harmonizing the design of the product with the way the purchaser

organization does business; from inside their own product development organization, these same ergonomists can be called upon to help harmonize their own organization with the product development process, with the manufacturing process, with the product distribution process, and/or with the product field support process.

The next step in the SE lifecycle process is product implementation; this includes iterative preproduction development of the product (in increasingly more refined form) and mass production or distribution of the product. The ergonomist can add significant value to both of these processes. In the pre-production stage, the ergonomist can provide a number of analytic evaluations of the product including heuristic analyses, managing expert reviews, and conducting laboratory-based usability analyses. As required by the FDA Quality System Regulation [17], test procedures that are appropriate for their intended use (validated test procedures that possess the appropriate sensitivity, specificity, and reliability), properly calibrated equipment, and tests that are statistically valid must be employed for usability studies. In the production phase, the ergonomist can assist in job redesign, the development of job aids, as well as recommendations on environmental and organizational issues that would enhance the productivity and job satisfaction of production personnel.

4.3. Compliance engineering

There exist a "hidden" set of changing laws, regulations, and standards (both national and international). They impose design, testing, implementation, and disposal constraints on the organization. Furthermore, they vary across industrial sectors and political boundaries, thus confounding the successful product development process. Compliance engineering involves the identification, applicability assessment, design impact, test design, and operation/disposal considerations required to conform to these constraints. Compliance engineering is an important source of requirements—constraints being the inverse of requirements.

There exist a large number of ergonomics standards; they address various aspects of the profession's activities and they are not generally well-known outside the profession. The ergonomist on the product development team plays a critical role in identifying, interpreting, and designing the product (e.g., the health information system) to conform to these constraints.

4.4. Reliability engineering

Safety (the absence of hazards) is a system property and but one aspect of reliability. Reliability implies proper functioning and safety is but one of the requirements that must be achieved for proper functioning. A corollary of this is that an unsafe system is an unreliable system [25].

One normally thinks of reliability engineering in terms of parts wearing out or undiagnosed software faults or failures. However, there is another dimension to the reliability equation—user reliability and use errors. Typically, non-ergonomist designers consider only the most obvious failure modes or well-known use errors. Ergonomists, by contrast, are trained to use analytical and laboratory techniques to discover the more subtle—but potentially more hazardous—use errors. With these same analytical and laboratory techniques, putative mitigations can be evaluated and the residual risks can be properly assessed.

Risk reduction is managed through risk identification, risk assessment, risk mitigation, and then *re-assessment* of residual risks. All members of the design team, including the ergonomist, utilize standard risk analytic techniques (e.g., fault tree analysis, failure mode effects and criticality analysis, or hazard and operability studies). However, the ergonomist begins not from an analysis of the mechanical or electronic parts or from an analysis of the program structure, but rather from a task and function analysis; the focus is *the interface between the device and the user*. Unlike the other members of the design team, the focus is on:

1. *hardware issues* (e.g., size, feel, color, and arrangement of physical controls and displays and the impact on their use with and without surgical gloves),
2. *software issues* (e.g., mental workload issues, logic of operations issues, training materials, etc.),
3. *environmental issues* (e.g., the crisis of a patient in cardiac arrest, the boredom and reduced vigilance at the end of a shift, light levels during day and night operations), and
4. *organizational issues* (e.g., purchaser organization administrative procedures for handling/using product and for scheduling work time, including multiple shifts, etc.).

At the end of each step in the SE lifecycle, it is essential to update the hazard analysis! The iterative hazard analysis plays a crucial role in SE and is a "gating function," permitting transition to the next step or looping back to the previous step. For products and processes that impact human health and safety, conducting iterative hazard analyses as decisions are made and modified throughout the development lifecycle provides an important mechanism for anticipating latent errors. Kossiakoff and Sweet [26] emphasize that "Reducing program risk is a continual process throughout the life cycle." Integral risk management activities are crucial from the FDA's perspective [27]. This is reiterated in ISO standard 14971:2000 [28]. From an ergonomics

perspective, hazards associated with the transition from "needs" to "requirements" include such items as whether all the requisite user populations have been properly identified and whether needs elicitation is statistically valid, so that it can be relied upon to properly represent the user populations. Ergonomically oriented hazards associated with the transition from "requirements" to "specifications" include such items as whether physical size constraints (based upon gender, nationality, etc.) are being adequately translated into mechanical engineering specifications. The proper formulation of use risk items, just as the proper formulation of requirements and engineering specifications, is a context-dependent process that is the domain of trained ergonomists. The proper formulation of use risks involves continual and intimate involvement of the ergonomist with the rest of the product design team.

4.5. Comparison with recent models

A number of lifecycle models have been published over the past two decades, with an emphasis on information systems and user interfaces. They do not comprise a comprehensive list of models and anything not *explicitly* stated in the published model was assumed to be absent for the purposes of this analysis. Each published model has been recast in the SE lifecycle framework (see Fig. 2); missing elements have been grayed out (Figs. 5–10). Gould and Lewis [29] emphasize iterative design with careful study of users and empirical measurements. Mantei and Teorey [30] closely follow the classical model, but omit the incremental hazard analyses, do not identify the testing process involved in verification of the design specifications and the CAPA process. Nielsen [31] also emphasizes iterative design and empirical testing cycles, careful study of the user and establishment of usability goals (requirements). Kreitzberg [32], May-

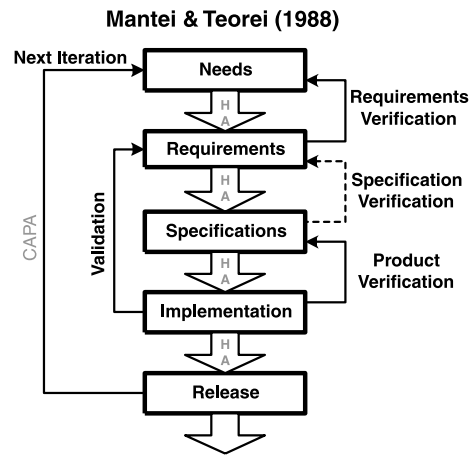

Fig. 6. Mantei and Teorei [30].

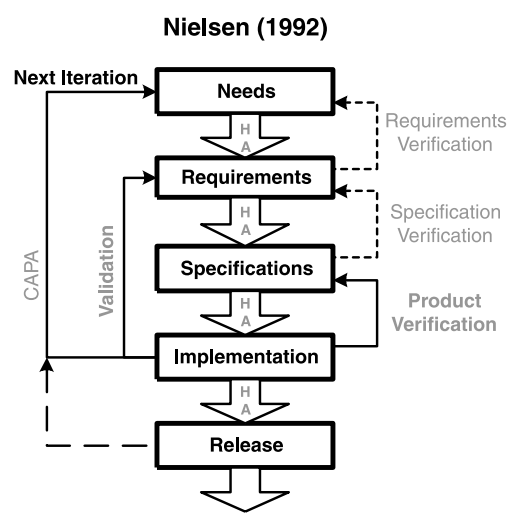

Fig. 7. Nielsen [31].

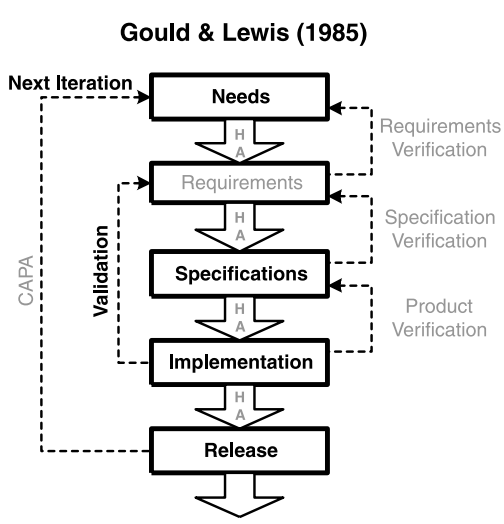

Fig. 5. Gould and Lewis [29].

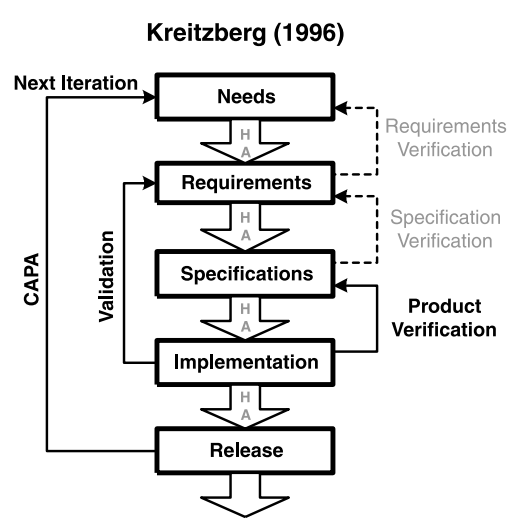

Fig. 8. Kreitzberg [32].

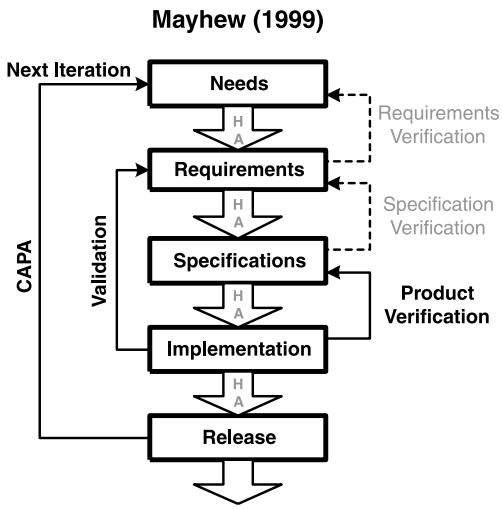

Fig. 9. Mayhew [33].

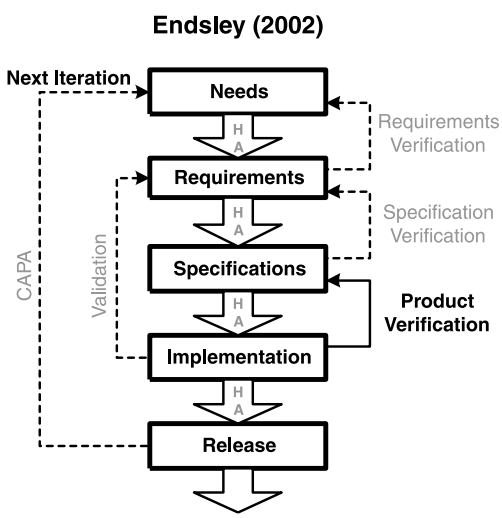

Fig. 10. Endsley [34].

hew [33], and Endsley [34] all emphasize iterative designs, emphasis on the user, and product verification.

We determine by inspection that these are all partial models of the classical systems engineering method. We were unable to find, in any of the published models, a justification (cost, schedule, or management) or benefit for eliminating the "grayed out" elements of the classical process. The absence of iterative risk analysis may be simply because these authors consider it part of the management process, rather than the development process. Nevertheless, iterative risk analysis is an essential part of reliability engineering and, as mentioned in the prior section, should be implemented at each stage in the development process.

5. Applications

Since 1964, the National Society of Professional Engineers, through its adoption of its Code of Ethics that "holds paramount the safety, health, and welfare of the public," has emphasized the consideration of safety, health, and welfare of humans involved with engineered systems [35]. SE provides a structured, systematic approach to risk reduction that is more cost-effective than ad hoc methods and maximizes the likelihood that design efforts will yield safe and effective products or processes. In considering how a SE approach can be applied to ergonomic problems in the design and development of health information systems, a number of practical matters become apparent. Issues that typically arise include the readiness of the organization to embrace a SE approach, the degree of formalization that is appropriate for a given project, what metrics one should use to characterize the human factors challenges that are inherent in a given system, and what software tools can be adopted to facilitate the SE engineering process.

As implied earlier, some health information systems are embedded in products that are presently regulated by the FDA as medical devices. As health information systems begin to have more profound effects on individual patient care (e.g., see the recommendations of the Institute of Medicine, [4,7]), the FDA may also begin to consider them as medical devices. It is likely that as networked database systems are shown to have patient safety implications (e.g., blood bank systems are treated in this manner at present), they will become subject to such regulatory processes. For the developers of such systems, there will be little choice as to whether to implement an SE process. It will be mandated by the inculcation of SE in the FDA's Quality System Regulation [17].

Another important consideration for health system information providers, of course, is the cost of implementing this SE process. But a complete and correct economic analysis requires that these costs of implementation must also be viewed in the context of the potential cost of NOT following a systematic SE approach, i.e., the costs to the organization if something goes wrong in the production, use, or disposal of the system. One need not look far for dramatic examples of what can go wrong when systems are developed and fielded without an eye towards systems engineering, and particularly hazard analysis [5,36].

5.1. Some accident scenarios and how they could have been avoided

Often the conditions that lead to a system failure that is attributed to human error can be traced to designs that did not take account of the full range of operating conditions, did not adequately consider human cogni-

tive or physical limitations, did not fully consider the extent to which communications among teammates might breakdown under stress, or did not provide appropriate feedback to the individuals or organizations involved. By following a SE approach throughout the product development lifecycle, such oversights can be minimized or avoided.

5.2. Over-exposure to radiation therapy: an older incident

In the chapter, "Set Phasers on Stun," Casey describes the 1986 case of a patient who was accidentally exposed to a massive, and ultimately lethal, dose of radiation during treatment for a tumor on his shoulder [36]. The technician using the radiation therapy machine incorrectly typed an "X," calling for the maximum power, "X-ray" mode, realized her mistake, and quickly corrected it by typing an "up arrow" and "e," for "electron beam" mode. Unfortunately, this sequence of keystrokes occurred more quickly than the designers of the device had anticipated, leaving the device in the "X-ray" mode, despite the fact that the display indicated that it had been switched to "electron beam" mode. When the beam was subsequently activated, the patient received a dose of radiation that was 125 times the prescribed dose. To make matters worse, the radiation therapy device then reverted to a "malfunction" mode which displayed a message to the technician suggesting that no radiation had been delivered. She then re-activated the machine twice, repeating the overdose.

Obviously, the design process for the radiation therapy device that led to this patient's death was flawed. It did not take into account the capability of the technician to enter the sequence of keystrokes to change modes as quickly as she did. It apparently did not anticipate the likelihood that technicians would need to execute this sequence of keystrokes, despite the fact that it seemingly represented a typical cognitive self-correction. Moreover, the machine reverted to an error mode that presented a misleading message, which the technician interpreted as indicating that no radiation had yet been delivered.

These flaws could have been avoided at several junctures in a systematic VE process. In the Needs Assessment and Requirements setting process, the likelihood of the technician detecting a mental lapse and correcting herself should have been anticipated. Likewise, the speed with which human operators, having such intent, could enter the keystrokes to change modes should also have been taken into account in designing the mechanics and messaging built into the machine. The failure to design for this sequence of keystrokes should have been picked up during hazard analyses that explored the extent to which requirements had been translated into specifications or the extent to which specifications had been successfully implemented. Such hazard analyses should also have pointed out the potentially disastrous effects of the error message that prompted the technician to reactivate the device and repeat the overdose.

5.3. Over-exposure to radiation therapy: a more recent incident

In a 2001 incident, which occurred in an oncology treatment center in Panama, 28 patients were overexposed during radiological therapy and 5 died. The investigation [37] concluded that the problem arose in the misuse of a treatment planning system. The system required that user enter data on the spatial co-ordinates of shielding blocks used to protect healthy tissue during radiotherapy and that these shielding blocks be entered into the system one block at a time, following a certain sequence and subject to a limitation on the number of blocks (four or fewer). One of the radiation oncologists decided to add a fifth block, and the physicist in charge devised a new method to overcome the four block limitation. Instead of digitizing the blocks individually, i.e., one block at a time, the staff members entered the contours as one complex block, with a first loop following the inner boundaries of the block, then with a second loop following their outer boundaries. This method of using the treatment planning software was neither recommended nor forbidden by the system documentation. Moreover, the display presented to the user suggested that the shielding contours had been implemented as intended. However, the underlying algorithms (it was later determined) were dependent on the direction in which the user drew the contours. If the second outer loop was drawn in the opposite direction to the inner one, the computer calculated a correct treatment time. But if the outer loop was drawn in the same direction as the inner one, the computer accepted the data, but calculated a wrong treatment time, doubling the dose to the patient. Retrospective investigations confirmed that the staff member had performed the latter procedure, that this use of the system was inappropriate, but that the system documentation was confusing and incomplete, the display was misleading, and the algorithm should have been more robust.

This unfortunate incident emphasizes that unforeseen usage patterns in health software can be lethal and again highlights the need for systematic hazard analysis. System software should prevent the misinformed, but well-intentioned user, from creating calculations that could deliver inappropriate and unintended outputs. Displays should accurately reflect the input conditions, and to the extent possible, provide insight into the underlying algorithms being invoked. System documentation should be complete, accurate, and readily usable. While it may not be possible to foresee all such inappro-

priate usage, the system can, by and large, be engineered to protect against such misapplications. Task analyses and user involvement in the formative design process should reveal design pitfalls. User testing, of both the software and documentation, under realistic operational conditions should highlight possible misconceptions. Hazard analyses should catch and allow correction of any previously unforeseen design shortfalls.

5.4. Implementing SE to avoid such problems

5.4.1. Organizational maturity

Different organizations can be viewed as being at different levels of maturity regarding the implementation of SE—inactive (i.e., not conducting SE approaches to speak of), reactive, interactive, and proactive. Regulated industries, such as the medical device industry, are required by the FDA to be proactive in implementing these approaches. Depending on the type of product, there may be various stakeholders whose needs should be addressed in the engineering process—ergonomists often focus, as well they should, on the needs of the users (patients, clinicians, and operators), but one sometimes also needs to take account of the ergonomic issues related to the role of product managers, designers and developers, producers/assemblers, maintainers, and disposers (Fig. 3). There are often overlapping interests among these various constituencies, but on occasion trade-offs and compromises must be made. A systematic SE approach should be explicit in exploring the cost-benefit implications of any such trade-offs and in documenting the choices made.

5.4.2. Degree of formalization that is appropriate

The degree of formalization involved in such project documentation, and the methods and tools adopted to facilitate the process, can be tailored to the criticality and complexity of the system with which one is dealing. There is no need to produce sophisticated test procedures and electronic databases of test results, when the level of detail in the requirements, specifications, hazard analyses, and test results are such that they could be handled by checklists or spreadsheets. However, when criticality is high (e.g., patient safety is at stake or there may be toxic impacts on the environment) and/or complexity is high (e.g., as dictated by project size, time constraints, or project team distribution), then a higher degree of formalization, and more sophisticated tools to facilitate the process, are in order. The formalization positioning diagram, illustrated in Fig. 11, attempts to convey this relationship. Minimum formalization may entail only paper or electronic checklists, spreadsheets, or flow charts. Moderate degrees of formalization may entail databases of requirements, test methods and parameters, test results, traceability matrices, and/or attribute matrices. Maximum formalization efforts, given the high degree of criticality and complexity involved in the system under scrutiny, almost surely require software engineering tools to manage and track the SE process and test results.

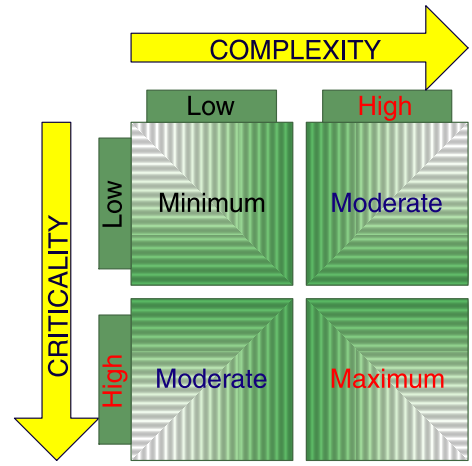

Fig. 11. Degree of formalization.

5.4.3. Measurement for SE

One overriding principle in the application of SE is the need to be as quantitative as possible. In considering ergonomic issues, one might think that we are hampered by the inherent unreliability of measurements that characterize the human element in system performance. However, a convincing case can be made for the fact that measurement issues in ergonomics are not so different from the measurement issues in physical systems, by which hardware and software SE is conducted. The range and degree of precision may be different, but the basic principles of measurement still hold. There are many measures by which users and user behavior can be characterized:

- Behavioral/Performance indices
- Accuracy (e.g., success rate; detection rate; and tracking deviations)
- Incidence of error types (omission, commission, etc.)
- Time on task; response time
- Ratings of subjective dimensions (e.g., user satisfaction, workload, stress, and fatigue)
- Anthropometric indices (e.g., height, weight, and length)
- Biomechanical indices (e.g., force, pressure, and angular velocity)
- Physiological indices (e.g., heart rate, pupil dilation, and eye blink frequency).

Measures such as these can help cast ergonomic problems, and possible interventions, in a SE framework. One cannot manage what one cannot control. One cannot control what one cannot measure. One cannot mea-

sure what one cannot operationally define. And one cannot define what one does not know about. A SE framework will help define and track the design issues that need to be considered. In so doing, it encourages measurement of design parameters and human performance with the system and ultimately controls the risk that is entailed in system use.

5.5. Available tools that can be adapted for SE

Fortunately, there are now a wide variety of software tools that are available to help manage the practical implementation of a SE process. Available tools that can be adapted for ergonomic SE efforts can be categorized as follows:

- Hazard analysis tools
- Requirements engineering tools
- Compliance engineering tools
- Reliability engineering tools
- CAPA Tools.

5.6. Hazard analysis and tools

Key to the SE process is the management of use-related hazards and the consequent delineation of system requirements. Use-related hazards may stem from any of the following aspects of a system:

- Used in unanticipated ways
- Used in anticipated ways, but inadequately controlled for
- Requires physical, perceptual, or cognitive abilities that exceed those of particular users
- Inconsistent with user expectations or intuitions
- Environment affects operation and effect is not recognized or understood by the user
- User's physical, perceptual, or cognitive capacities are 'exceeded, when in a particular environment.

Some hazard analysis tools that can be adapted for use in characterizing ergonomic hazards are the following:

- Dyadem International, Ltd. Tools:
  - FMEA www.dyadem.com/products/fmea/index.htm
  - HazOp www.dyadem.com/products/pha-pro/index.htm
- Relex Software:
  - FMEA www.relexsoftware.com/products/fmeafmeca.asp
  - FTA www.relexsoftware.com/products/faulttree.asp.

Too often, only single point failures are considered. Of critical importance in the hazard analysis is the consideration of "multi-point" failures that will interact to "defeat, bypass, or disable our safety devices" [38]. Perrow points out that such so-called "system errors" may be reduced by reducing system complexity and coupling [38]. A similar warning is put forth by Reason [39], who states that "it leaves systems prey to the one hazard for which there is no technological remedy: the insidious concatenation of latent human failures that are an inevitable part of any large organization." Only by a carefully managed, structured, systematic human-centered systems engineering approach can we decrease the complexity of system behavior and identify many (though clearly not all) latent errors.

5.7. Requirements engineering and tools

Requirements engineering, in the context of ergonomics, involves determining human needs (both those of the patient and those of the system user, who might be a caregiver, a technician, or for that matter the patient himself), deciding which needs will be addressed, documenting the desired external behavior of the system (i.e., identifying features and associated requirements), quantifying these requirements, verifying (i.e., testing) these requirements, and eventually updating the requirements for the next iteration of design.

Well-formed requirements have the following characteristics—they lack ambiguity, are complete, are consistent, can be traced to their origins, are not tied to specific design solutions, are verifiable and testable, can be enumerated and categorized, and have attributes that can be identified and assigned. A systematic requirements engineering process, as it applies to the ergonomics of health information systems, involves determining user needs, deciding which needs will be addressed, writing down the desired external behavior of the system (identifying features and associated requirements), quantifying those requirements, then testing and verifying/validating that those requirements are actually met, and if necessary updating those requirements for the next iteration of design. There are many tools available to facilitate the requirements engineering process. Information about such requirements engineering tools is available at the following sources:

- A Survey of Requirements Engineering Tools (www.volere.co.uk/tools.htm)
- INCOSE Requirements Engineering Tools Taxonomy (www.incose.org) "Quick Links."

5.8. Compliance engineering and tools

As requirements and their resulting specifications are verified, there is the need to document and track

the compliance checking process. This can become quite complex as different measurement standards are invoked, perhaps different licensing requirements in different jurisdictions are brought to bear, and the inspection process and its outcomes are documented. A set of tools (License 2000, MYLicense, Mcheck) that are customized for tracking compliance in the context of government licensing processes are the following:

- System Automation Corporation tools (www.systemautomation.com/products.htm).

5.9. Reliability engineering and tools

There are many aspects in which a system can fail. The loci of failures can be at the level of the hardware, the software, the human operator, or at the system level (i.e., involving the interactions among these various levels). In attempting to quantify system reliability, there may often be both a prospective and a retrospective aspect to be considered. One might sample system performance based on an operator's usage of a prototype in an attempt to characterize the probability or risk of failure. One might also document the performance of previous versions of a system in order to determine, in practice, how a system or several of its components functioned historically. Assessing the reliability of human performance may seem to be a daunting task, fraught with unreliability in the measurement process itself; however, the metrics alluded to above can be applied with the same principles and data collection formalizations as those applied to physical aspects of a system.

The feedback loops in the SE process involve verifying that the system as designed, and eventually as built and deployed, meets the requirements and specifications that have been defined for it. This involves testing and measurement, and as applied to ergonomic issues, this implies the need to observe system use in a realistic context of care. A variety of methods can be brought to bear here, but they have in common a reliance on collecting data from representative users as they make use of a prototype system under realistic conditions. Several of the software tools that have been developed to facilitate the characterization of system reliability, and which can be adapted for measuring the effectiveness and efficiency of human performance in operating complex systems, are the following:

- Relex Software Corp. Tool Suite (www.relexsoftware.com/products/index.asp)
- ReliaSoft's Reliability Growth Analysis Tool (http://rg.reliasoft.com/)
- Item Software, Inc. Tool Kit (www.itemsoft.com/itoolkit.html).

5.10. CAPA tools

Even after the deployment of any complex system, flaws will be found. There need to be systematic ways of capturing and correcting these flaws and preventing their recurrence. An important feedback loop in the SE process involves Corrective and Preventative Actions (CAPA). CAPA pertain to the next iteration of product design and development. There are several software tools that have been developed to facilitate the CAPA stage of the engineering process:

- Relsys, Inc. EasyTrak (www.relsys-inc.com/products/easy_trak/overview.asp)
- Pilgrim Software, Inc. SmartCAPA (www.pilgrim-software. com)
- ReliaSoft's FRACAS++ (www.reliasoft.com/enterprise/fracas.htm).

5.11. Cost justification

SE processes incorporating human factors knowledge and expertise will be seen as valuable to the extent that they save money for organizations. While many engineering processes in the biomedical and healthcare arenas are driven by licensing and regulatory demands, and fear of litigation, there can also be other means for justifying the costs of ergonomics in a SE framework. Growing data suggest that the application of human factors knowledge have the effect of decreasing development time and costs, increasing productivity and efficiency, decreasing the cost of operations, and increasing sales and revenues [40]. The benefit of using a structured, systematic approach is well-known in the systems engineering arena (e.g., [18] and [41]). Decreases in development costs result from fewer design changes late in the development process. Fewer retrofits after product release, just-in-time supply of parts and services, and focusing and coordinating the efforts of the design team also increase profit margins. Decreases in the cost of operations stem from fewer catastrophic failures, increased productivity, decreased need for training, and decreased costs with timely maintenance and support. These have been well-recognized in the systems engineering arena for decades.

6. Macroergonomics

While it may be clear from the foregoing that ergonomists can make valuable contributions to advancing the

structured, systematic consideration of human issues in the development of products and processes, their full participation as *equal partners* in the endeavor is often thwarted by organizational issues. Often, the fundamental impediment to serious and detailed consideration of human issues, during product or process development and deployment, is organizational design—the historical structures and functions by which the organization has previously succeeded, or at least, survived. There is a natural human reluctance (often termed bureaucratic inertia) to modifying what has worked in the past—even if it has not worked well. And yet this is one of the principal domains of ergonomics. How does one approach the requisite organizational change? We believe that a logical approach is to employ essentially the same SE model previously described for microergonomic involvement in the development of products and process! It is, after all, a general problem-solving method that is not domain specific. Here, though, instead of developing new physical products or production processes, we will be developing new work structures and processes (Fig. 12).

The organizational design activities required by SE elucidate clearly the "steps usually carried out in an over-lapping, iterative, and non-linear manner" to design an organization's work system structures and processes [42]. The adoption of the SE process helps avoid the standard pitfalls of organizational design, which Hendrick [42] identifies as (a) human interface design for *already designed* systems; (b) the non-human-centered or the "left-over" design approach; and (c) failure to consider and integrate the organization's socio-technical characteristics into the design of the work structures and processes.

Only the "range of disciplines" of the SE space will be transformed for a macroergonomic endeavor—from the microergonomic [*hardware–software–ergonomics–seller/purchaser economics*] to the macroergonomic [*management–operations–personnel–finance*]. This new range of disciplines reflects the elements essential for organizational change. The domain (requirements, compliance, and reliability) remains the same, as does the fundamental time horizon.

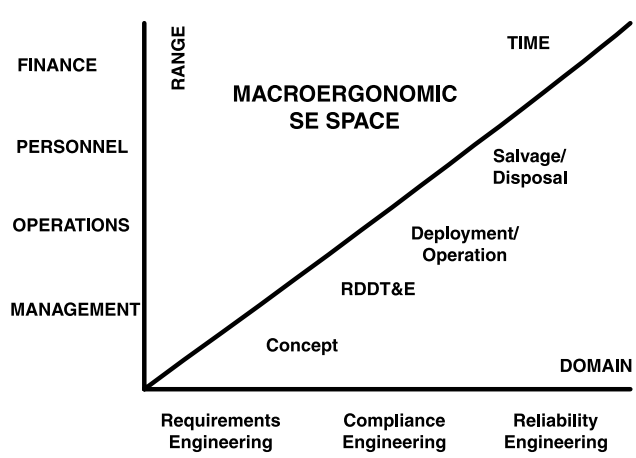

Fig. 12. Macroergonomic SE space.

From a lifecycle perspective, we begin with the determination and analysis of the organization's needs and wants (i.e., its objectives and goals for the work system). This puts the consideration of ergonomic criteria as early as possible [43]. We transform these into appropriate requirements and verify that the requirements (and constraints) conform to the organizational needs and wants that have been specifically selected for implementation.

These requirements are translated into organizational design specifications (managerial, operational, human resource, and financial specifications—remember it costs money to have additional people "sitting at the table"). Once these specifications have been verified against the requirements, the iterative process of implementing the requisite work structures and processes begins. Carayon [43] highlights the issue of work implementation in a high-pace, high-pressure environment. It is crucial to recognize that implementation of new or revised work structures must carefully consider the criticality and complexity of the processes. The requisite degree of formalization is, once again, depicted by Fig. 11. Once the implementation is verified against the specifications and validated against the requirements, the new work structures and processes are released. Post-deployment CAPA studies, in terms of managerial, operational, personnel, and financial issues, drive the next organizational design iteration. The timescale and approach are fundamentally the same as for development of any other system. Salvage and disposal of particular processes and work breakdown structures, as the needs of the organization change, are no different conceptually than salvage or disposal of tangible assets.

Not all organizational structures are directly susceptible to this approach. For example, spontaneously self-organizing teams and ad-hoc project teams coalesce so rapidly that we can have little control over their ephemeral development. However, we do have control over the environment in which they arise (specifying their expected external behavior and any constraints imposed upon them) and the structured, systematic development of that working environment is a fundamental responsibility of management.

As with microergonomic applications, applying the SE method to organizational issues has the profound benefit of making the detailed decision-making processes *structured*, *systematic*, and *transparent*.

7. Conclusions

The SE paradigm is a "cradle-to-grave," structured, systematic approach to system risk reduction in product

or process development. It is based upon the triumvirate of requirements engineering, compliance engineering, and reliability engineering; it applies to the microergonomic range of hardware engineering, software engineering, human factors engineering, and seller/purchaser economics. Furthermore, the SE paradigm can be applied to macroergonomic endeavors, when it is appropriate to effect organizational change.

The SE method clearly elucidates the important role that ergonomics should play in product or process development. It provides a framework of incorporating human factors engineering knowledge. It clarifies for project managers the complementary roles of hardware, software, and human factors engineers. Finally, it justifies the continual involvement of ergonomists throughout the project lifecycle—rather than just at the beginning or end of the project!

Based upon a graphical analysis, we observe that various recently published lifecycle models may be viewed as subsets of the classical SE lifecycle model. In the aggregate, these models contain essentially all the elements of the classical model (except for explicit inclusion of the iterative incremental hazard analyses). While the failure to consider the full SE model in system development efforts is not limited to any particular application domain, the consequences of doing so may be particularly important in health information systems because of their safety criticality.

Thus, the health information systems domain can benefit from the use of the classical systems engineering method, whose utility has been demonstrated repeatedly in other arenas over the past half century. Furthermore, since the SE method described is that detailed by the FDA (that specifically requires inclusion of human factors considerations), it will simplify compliance of health information systems that may come under the regulatory purview of the FDA in the future.

Moreover, the ergonomics profession can benefit from the application of the classical SE approach as well as contributing to an organization's product/process development effort utilizing the SE model. The SE model provides a paradigm for enabling a structured, systematic human-centered design approach, incorporating ergonomics knowledge and allowing ergonomists to contribute throughout the system development process.

References

[1] Chapanis A. Human factors in system engineering. New York: -Wiley; 1996. p. 10.

[2] Ergonomist Formation Model, Board of Certification of Professional Ergonomists, Available from: http://bcpe.org/info/detail.asp?RecordID=3. Accessed November 8, 2004.

[3] Bogner MS. Human error in medicine. Hillsdale, NJ: Lawrence Erlbaum; 1994.

[4] Kohn LT, Corrigan JM, Donaldson MS, editors. To err is human: building a safer health system. Washington, DC: National Academies Press; 2000.

[5] Bogner MS. Misadventures in health care: inside stories. Hillsdale, NJ: Lawrence Erlbaum; 2004.

[6] FDA Human Factors Implications of the New GMP Rule Overall Requirements of the New Quality System Regulation. Available from: http://www.fda.gov/cdrh/humfac/hufacimp.html. Accessed November 8, 2004.

[7] Aspden P, Corrigan JM, Wolcott J, Erickson SM, editors. Patient safety: achieving a new standard of care. Washington, DC: National Academies Press; 2004.

[8] Buede DM. The engineering design of systems: models and methods. New York: Wiley; 2000. p. 6.

[9] Hall AD. Systems engineering from an engineering viewpoint. IEEE Trans Syst Sci Cybern 1965;SSC-1:4–8.

[10] Sage AP. Systems engineering: methodology and applications. New York: IEEE Press; 1977. p. 11.

[11] Engineering Management, Military Standard 499A (USAF). Washington, DC: United States Department of Defense. 1974.

[12] Blanchard BS. System engineering management. 3rd ed. New York: Wiley; 2004. p. 18.

[13] Nadler G. Systems methodology and design. IEEE Trans Syst Man Cybernet 1985;15(6):685–97.

[14] Sage AP. System design for human interaction. New York: IEEE Press; 1987.

[15] Chapanis A. Human factors in system engineering. New York: -Wiley; 1996. p. 38–9.

[16] Buede DM. The engineering design of systems: models and methods. New York: Wiley; 2000. p. 18–9.

[17] FDA Quality System Regulation, US FDA 21 CFR Part 820, October 1996.

[18] Blanchard BS. System engineering management. 3rd ed. New York: Wiley; 2004. p. 5–7.

[19] Blanchard BS. System engineering management. 3rd ed. New York: Wiley; 2004. p. 168.

[20] Kossiakoff A, Sweet W. Systems engineering principles and practice. New York: Wiley; 2003. p. 98–100.

[21] Chapanis A. Human factors in system engineering. New York: -Wiley; 1996. p. 291–5.

[22] Human-centered Design Processes for Interactive Systems. International Standards Organization, ISO Standard 13407:1999.

[23] Martin JN. Systems engineering guidebook: a process for developing systems and products. New York: CRC Press; 1996. p. 4.

[24] Eisner H. Essentials of project and systems engineering management. New York: Wiley; 1997. p. 165.

[25] Musa JD. Software reliability engineering. New York: McGraw Hill; 1999. p. 19.

[26] Kossiakoff A, Sweet W. Systems engineering principles and practice. New York: Wiley; 2003. p. 98.

[27] Herrmann DS, Zier DA, Using IEC 601-1-4 to satisfy FDA software guidance requirements, Medical Device and Diagnostic Industry Magazine; 1995, p. 104.

[28] Application of Risk Management to Medical Devices, International Standards Organization, ISO Standard 14971, 2000.

[29] Gould JD, Lewis C. Designing for usability: key principles and what designers think. Commun ACM 1985;28(3):300–11.

[30] Mantei MM, Teorey TJ. Cost/Benefit analysis for incorporating human factors in the software lifecycle. Commun ACM 1988;31(4):428–39.

[31] Nielsen J. The usability engineering life cycle. IEEE Comput 1992:12–22.

[32] Kreitzberg C. Logical user-centered interaction design methodology from cognetics corporation—cited in Shneiderman B. designing the user interface: strategies for more effective human–computer interactions. 3rd ed. Reading, MA: Addison-Wesley; 1998. p. 104–7.

[33] Mayhew DJ. The usability engineering lifecycle. San Diego: Academic Press; 1999.
[34] Endsley MR. From Cognitive Task Analysis to System Design: Designing to Support Situation Awareness, CTA Online Seminar, 2002.
[35] History of NSPE Code of Ethics for Engineers, National Society of Professional Engineers, Available from: http://www.nspe.org/ethics/eh1-codehist.asp. Accessed October 31, 2004 (NSPE members area).
[36] Casey S. Set phasers on stun and other true tales of design, technology, and human error. Santa Barbara, CA: Aegean Publishing Company; 1993.
[37] Gage D, McCormick J, We Did Nothing Wrong: Why Software Quality Matters, Case 109 A Dissection, Baseline, 2004, p. 4–21. Available from: http://common.ziffdavisinternet.com/download/0/2529/Baseline0304-DissectionNEW.pdf. Accessed November 11, 2004.
[38] Perrow C. Complex organizations: a critical essay. 3rd ed. New York: McGraw-Hill; 1986. p. 148–9.
[39] Reason J. Human error. New York: Cambridge University Press; 1990. p. 250.
[40] Bias RG, Mayhew DJ, editors. Cost-justifying usability. San Diego: Academic Press; 1994.
[41] Kossiakoff A, Sweet W. Systems engineering principles and practice. New York: Wiley; 2003. p. 16–9.
[42] Hendrick HW. Organizational design and macroergonomics. In: Salvendy G, editor. Handbook of human factors and ergonomics. New York: Wiley-Interscience; 1997.
[43] Carayon P. Macroergonomics in Quality Care and Patient Safety, in Human Factors in Organizational Design and Management—VII, Luzak H, Zink KJ, editors. Santa Monica, CA: IEA Press; 2003, p. 21-34. Available from: http://www.wpsi.org/media/documents/pdf/2003_Highlights/WI_Forum_Carayon_C.pdf Accessed November 9, 2004.