

Unifying Human Centered Design and Systems Engineering for Human Systems Integration

Guy A. Boy¹ and Jennifer McGovern Narkevicius²

Abstract. Despite the holistic approach of systems engineering (SE), systems still fail, and sometimes spectacularly. Requirements, solutions and the world constantly evolve and are very difficult to keep current. SE requires more flexibility and new approaches to SE have to be developed to include creativity as an integral part and where the functions of people and technology are appropriately allocated within our highly interconnected complex organizations. Instead of disregarding complexity because it is too difficult to handle, we should take advantage of it, discovering behavioral attractors and the emerging properties that it generates. Human-centered design (HCD) provides the creativity factor that SE lacks. It promotes modeling and simulation from the early stages of design and throughout the life cycle of a product. Unifying HCD and SE will shape appropriate human-systems integration (HSI) and produce successful systems.

1. Introduction

The International Council on Systems Engineering (INCOSE) defines systems engineering (SE) as “an interdisciplinary approach and means to enable the realization of successful systems.” The SE approach includes early definition of customer needs, documentation of the desired functionality and the technical requirements before design activities are begun. SE spans all stages of the life cycle of a product including design, manufacturing, operations, training and support, disposal and costs (Haskins, 2011). Even with this broad approach to system development, there remains much to improve to take into account and

¹ University Professor, Director of the Human-Centered Design Institute, Florida Institute of Technology, and IPA Chief Scientist for Human-Centered Design at NASA Kennedy Space Center.

² Jenius LLC, and Chair of the Human-Systems Integration working group of INCOSE.

define people's needs wholly. Achieving human-centered SE requires a mindset that differs from current technology-centered and finance-driven practices. We observe consistent program and project failures despite adherence to the very holistic SE approach. These failures range from small and accommodatable (e.g., annoying "quirks" in handheld devices) to the truly spectacular (the failure of a power grid). The source of these failures can commonly be found in inadequate requirements that are technology focused, poorly specified, unmeasurable, and unachievable. The failure to incorporate a human-centered focus also results in the lack of these requirements driven by organizations and people, which are the most flexible but unrepresented interfaces in the system. SE methods and tools impose rigid processes that end in failure because our technology-based world evolves very quickly making flexibility in solutions essential.

Human-centered design (HCD) has developed across diverse technical disciplines drawing on human-computer interaction, artificial intelligence, human-machine systems, social and cognitive sciences. It is based on the same objective as SE, that is developing a holistic approach to support fuller integration of Technology, Organizations and People but HCD brings different concepts and terminology (*cf.* the TOP model: Boy, 2013). This broader, trans-disciplinary language supports a new, more comprehensive approach to systems design. In HCD, "systems" are commonly denoted as "agents" which can be people or software-based artifacts. Agents are functionally defined as displaying cognitive functions, defined by their roles, the contexts of validity and supporting resources (Boy, 1998). The roles can be expressed in terms of objectives, goals and/or purposes. Minsky (1985) defined an agent as a "society of agents" and by analogy, a cognitive function is a society of cognitive functions, which can eventually be distributed among different agents.

It is interesting to compare this HCD definition of an agent with the SE definition of a system. INCOSE Systems Engineering Handbook defines a system as an "integrated set of elements that accomplish a defined objective" (Haskins, 2011). Note the correspondence of Minsky's definition of an agent to the current definition of SE community's System of Systems (SoS) (Clark, 2009). The association of the concepts of systems and agents highlights the parallel paths of HCD and SE. SE processes include technology and people. This approach takes the functionality of people into account as a central part of systems. HCD includes agents (both the cognitive functions of people and technology). It is time to merge these two approaches to achieve a better definition of Human-System Integration (HSI).

HSI complexity can be analyzed by explaining the emergence of (cognitive) functions that arise from agent activity. This expansion of agents results in a need for function allocation approaches that supersede the classical MABA-MABA³ model (Fitts, 1951), where functions are not only allocated deliberately, but as they emerge from agents' activities. Our socio-technical world is dynamic and the concept of static tasks should also be superseded by the concept

³ Men Are Better At - Machine Are Better At.

of dynamic activities. A task is what we prescribe; an activity is what we effectively do. Activities are characterized by their variability and should be modeled by a dynamic non-linear model, not as signal and noise. This is where the difficulty lies - systems are no longer only mechanically complicated, they are highly interconnected and, therefore context-dependent and complex. People generally handle variability well, engineered systems cannot. Engineered systems are programmed (handling procedures only); people are flexible and creative. Engineered systems are excellent at deduction; people are unique at handling induction (Harris et al, 1993) and, more importantly, abduction (Peirce, 1958).

Conventional linear methods and techniques only work in the short term and are not useful for handling the complexity of HSI. It is imperative to develop and use methods and techniques built on complexity science that study non-linear dynamical systems, attractors, bifurcations, fractals, catastrophes, and more. Instead of avoiding complexity and working in the artificial world of the Gaussian "bell" curve, where we remove the parts that we do not understand or do not fit our hypotheses or a priori conclusion, we need to model the overall complexity of HSI and benefit from it. Design can only succeed by restoring the necessary flexibility of control, management and accountability of systems being designed and developed today. It is essential to understand why SE fails, why context matters, why HSI requires complexity analysis, and why HCD generates a path directly toward HSI.

2. Why does systems engineering fail?

Software is embedded in almost all systems. Until recently, the concept of machine was mainly associated with hardware and mechanical entities. Engineering cars was the kingdom of mechanical engineers. Electrical and computer engineering have progressively penetrated this industrial sector and current cars are full of wires and electronic circuits. It is no longer possible to repair a car without complex diagnostic systems. The next step, represented by the Google Car, is drastic. It promotes software engineering, human-computer interaction and artificial intelligence to lead the design of the entire car. This approach is backward - instead of putting electronics into a thermal engine and a car body, wheels are stuck onto a computer! We are designing tangible objects in which software will make itself as invisible as possible, working undetectably supporting the various tasks people have to perform.

Large industrial programs require organization. This significantly contributed to the development of SE. However, industrial organizations are still designed with hardware in mind. Software is still often considered as an add-on. Hardware engineering and software engineering should be integrated in a human-centered way (i.e., from purpose to means and not the other way around). This organization is essential to assure that the engineering work is supported and consistent, allowing good work by both hardware and software engineer-

ing. It is important to make a distinction between design and manufacturing. When design is well done, manufacturing can be easily accomplished using current SE approaches. However, when design is rushed or pressured by outside influences (including local politics or program management goals) the SE processes are aborted or abandoned resulting in “patches” and constant re-engineering of products. We then need to better use creative, flexible and effective software engineers cooperating with human-centered designers to do conceptual models and adapted human-in-the-loop simulations that lead to effective requirements for manufacturing.

Interconnectivity grows exponentially. Software is no longer only embedded locally; it enables interconnections among a variety of systems considered as agents. Internet is a great example of such interconnectivity. People are connected to other people and systems, anywhere and anytime, synchronously and asynchronously. Such interconnectivity creates communication with no delays; but new kinds of behaviors and properties emerge which cause new kinds of problems. This is why organizational models are so important. These models can be descriptive, prescriptive or predictive, and should be developed in that order. As a first step, they need to describe the various outcomes of SE use in industry and governmental agencies. Descriptive models require the development of appropriate syntax (e.g., nodes and links) and semantics (e.g., contents and meaning of these nodes and links). The military organizational model or architectural framework has been in use for a long time. It is based on pyramidal hierarchies with primarily vertical information flow. This model is still in use in most large organizations. However, in these organization information runs transversally (via mobile phone, email, chat, and other typically internet based communication). Several new types of communities have emerged from the use of this growing interconnectivity including communities of practice organized through social media. Such interconnectivity increases reactivity and may cause the emergence of events that would not have been possible before.

These emerging organizational models in a highly interconnected world must be analyzed. The Orchestra model (Boy, 2009 and 2013) was proposed to describe workers’ evolution from old-time soldiers (i.e., executants) to a new model, musicians (creative, flexible experts). The main difference between executants and experts is their level of autonomy. The more agents become autonomous, the more coordination rules they will need to interact safely, efficiently and comfortably in society. The composers of the Orchestra model are needed to create, coordinate and fine-tune the musicians’ scores. This is the first level of coordination, prescriptive coordination, the task level. During a performance, the conductor is required to coordinate musicians in real-time. This is the second level, effective coordination, is the activity level.

Rapid and constant socio-technical evolution. Our socio-technical world is changing fast and drastically and interconnectivity is the major contributor. Flexibility is required to enable the necessary rapid adaptation. Three main issues continue to be relevant: complacency, situation awareness and decision-

making. Automation is a contributing cause of complacency. Many systems are highly automated and human operators do not have much to do in nominal situations. Consequently, they do nothing or they do something else. In either case, they are barely in the control loop of the process they are supposed to monitor and manage. Issues arise when unanticipated events occur, when human operators have little time to jump back into the control loop, often leading to major issues. Recent commercial aircraft accidents have been attributed to this problem. To avoid this in systems design we need to look at the bigger picture of Technology, Organizations and People (the TOP model) to answer this question (Boy, 2012). The answer is not necessarily technological; it can be organizational, training, change of practice, or some combination of these.

This may be the first time in the history of humanity that remembering things is not the best way to keep them handy almost anytime, anywhere. The Internet gets the information that we want in a few clicks. However, having information handy does not guarantee that we understand it - the meaning is crucial (Boy, 2013). Pilots may face all kinds of relevant and timely information, but if they do not understand it in context, it is neither useful nor usable. This is why situation awareness progressively emerged as a fundamental concept. The TOP model points to appropriate solutions to this problem.

Procedures and rules impose rigidity, thwarting the flexibility required to solve emergent problems. The unexpected is crucial today because of our highly procedural world (Boy, 2013). Procedures can be imposed organizationally or embedded in automated systems (machines) but when something unexpected occurs, solutions are not handy to make appropriate decisions, and socio-cognitive adaptation becomes an issue. In such situations, decision-making is handicapped by lack of situation awareness, educated and embodied practice of problem solving and risk taking (Boy & Brachet, 2010).

3. Context matters

People issues. SE provides a framework that is often context-free. However, every project or program involves many different contexts that often lead to ad-hoc planning and operations, and adherence to SE procedures may establish roadblocks. Even if SE processes work well, they must be articulated by people. These people need to understand SE, organizations, contexts, human resource management and the space between these where major social issues emerge when SE processes are very complex. One of the main flaws in SE is to think that people are systems in the same way machines are. People are far more complex than engineered systems (i.e., artifacts). In addition, when people interact with simple artifacts, complexity emerges from the various possible usages of artifacts and other interactions which are difficult to anticipate exhaustively.

People have creativity, complexity, flexibility, inductive and abductive cognition. Modern artifacts equipped with sophisticated software can also display

complex cognition. People easily forget what they have to do; well-programmed software does not. People are unique inventors and creative creatures; machines are not. The set of capabilities and limitations that people and machines bring to the system solution are broad and complementary – if they are considered early in design. Context variations are a natural variability that needs to be modeled and understood to generate appropriate decisions and actions to develop and design successful systems.

People have their own motivations and systems engineers are no different. Work is always better done when workers are more autonomous and trusted. In a new world where the Orchestra model can be a reference, workers have to be able and agree to play the same symphony by articulating their own personal production. They alternatively need to be leaders and followers. For example, astronauts learn both leadership and *followership*. Depending on context, they may change their roles. This imposes excellent skills and deeper knowledge of the domain and organization where work must be accomplished. Motivation is a matter of respect and engagement. When SE processes are too rigid and constraining without apparent reasons, workers get less motivated because they feel that they are not respected. They may think that they are treated as chattel and robots (e.g., as a spreadsheet box with a dollar number in it). Engagement is also a crucial factor that embraces dedication, completion of work toward goal satisfaction, and pleasure to perform. It is important to include *design thinking* into engineering and management practices (Boy, 2013).

Organizational issues. Organizations are set up with rules and procedures. The global market economy and interconnectivity affords large organizations to involve contractors distributed across many countries which modifies work practices to include work needs to be dichotomized into parts that can be produced independently, and more integrated at the top level. This dichotomization involves standardization that cannot be static because of rapid and constant social and technical evolution. Top-level integration involves technical, social and organizational competence that may not be entirely present today in organizations that are almost exclusively led by finance managers and stakeholders. This absence of socio-technical leadership influences many factors including personnel motivation, delays and customers satisfaction.

Culture is another emerging property of socio-technical systems. Organizations develop their own culture which depends on a tradeoff between order and freedom. Organizational culture is autopoietic, that is, the culture is incorporated into systems being developed and in their usages (Maturana & Varela, 1980). An autopoietic system or organization is defined as a network of processes that can regenerate, transform or destroy components that interact among each other. We can see that for a given culture the creation and refinement of products will be inevitably based on this culture (e.g., if the culture is technology-centered, the product will also be technology-centered). A human-centered culture in systems design and engineering will result in a human-centered organization and human centered products.

Knowledge management. SE has become a way of thinking and a standardized international engineering practice. Process quality is improved continuously. Protocols are now systematic and there are prescriptive strategies for many situations. Incremental experience feedback feeds this process of improvement and this quality approach contributes to reduced variability. However, this is only true for well-specified problems and known situations. Unknown unknowns are still problematic.

Tools have been developed to support SE practice however this results in abundant generation of documents. Model-based SE was a response to this document-centered approach (Brown, 2011). For example, modeling tools are now used to capture and identify requirements, system element structure, traceability, verification, and configuration management. However, people are not taken into account as they should be.

Knowledge management is difficult because (discrete) documents cannot replace (continuous) human expertise. Storing information does not guarantee that people reading, hearing or interacting with it will understand the content, what the document makers wanted to report, or the knowledge that was meant to be captured. Documentation is contextual but context is difficult to capture. It requires document makers to have an external view of the (current) context and imagine the (future) context in which document readers will be.

4. Human-Systems Integration requires complexity analysis

Moving from Cartesian philosophy to complexity science. The Capacity Maturity Model (CMM) proposes a sequential set of processes (Paulk et al., 1995). Organizations become certified for the maturity of their processes but CMM does not guarantee product maturity nor does it promise that a product will be sustainable; maturity of practice is also necessary (Boy, 2005 and 2013).

Quality is a matter of tests of a technology within its environmental, organizational and individual use domain (the TOP model again). Tests are predicated on a Cartesian approach of successive parsing, moving away from the complexity of the whole to the simplicity of the atomic. Interconnectivity and rapid, constant socio-technical evolution, makes these tests more difficult. Linear local tests are not sufficient and sometimes not relevant. We need to investigate new methods for non-linear holistic tests. We need to look for emerging behavioral patterns, attractors and bifurcations, in the complexity science sense (Mitchell, 2008). We must move from Cartesian positivist methods, which are constraining and limiting toward phenomenological approaches.

Applying the Orchestra model to what we can observe in our industrial companies and governmental agencies, it is obvious that we have not moved from the military model to an emerging orchestra-like model. This older military model is formal and highly structured while the new emerging model is informal, more flexible, and leverages individual creativity as a strength that contributes to the performance of the whole. There is a disjunction between the

way organizations are still structured and the way people behave in them, which causes consistency and synchronization problems. Practices have already evolved within the Orchestra model, and they are imposed the old military model. The SE V model is very linear and deliberately imposed as a rigid prescriptive model. Unfortunately, not everything can be predicted and we do not have predictive models that can be accurate and robust enough to anticipate variations. The role of people is to handle these variations. Since such variations may happen anytime anywhere in the organization, the old model is very limited. Thinking of the organization in a positivist sense (i.e., the whole is equal to the sum of the parts) does not work anymore; complexity is too high and emergent properties require a gestalt. SE often fails because of the lack of a holistic model.

Discovering emerging phenomena. Understanding complex systems behavior starts by looking for emerging phenomena. For example, the use of new technology may typically generate surprises and face human operators with unexpected events. Instead of considering these events as noise, it is important to model the overall human-system activity by considering the parameters involved in observed emerging phenomena. Consequently, the concept of emergence is crucial in SE, especially when addressing SoS.

Stacey defines emergence as "... the production of global patterns of behavior by agents in a complex system interacting according to their own local rules of behavior, without intending the global patterns of behavior that come about. In emergence, global patterns cannot be predicted from the local rules of behavior that produce them. To put it another way, global patterns cannot be reduced to individual behavior." (1996).

It is interesting to watch the emerging behavior of termites. Each of these creatures looks very simple, but when they are working together they can produce amazing things including thermo-regulated galleries and habitats. We need to think in terms of phenomena, and we need to study them. Biologists and physicists have already developed models to study convection and diffusion phenomena. They use integrated partial differential equations to produce these emergent phenomena. Analog phenomena could be modeled for engineering processes.

More specifically, going from the old military model to an orchestra model will require the definition of new types of management. Composers, conductors and musicians in general know about emerging musical patterns. They can adjust their own contribution to participate in the making of these musical patterns. In highly interconnected, rapidly changing and diversity-rich complex systems, we necessarily need to study management issues. Performing SE locally in a homogeneous group of people is drastically different from doing it in a multi-cultural distributed environment.

Designing for flexibility. Choosing complexity science as a support for managing SE imposes a functional approach and a structural approach to system design. The main problem is that classical "a priori" function allocation does not

work since some functions, perhaps the most interesting ones, will emerge from operations. The utilization of knowledge, gleaned from experience to develop virtual prototypes to be iterated is essential. These prototypes can be easily designed, developed and tested with people in the loop which may result in discovering emergent functions. This is the only way to test human-system activity before anything has been manufactured.

Agent-based architectures should then be developed beyond the prototypes while incorporating the emerging features. These architectures are flexible functional networks that can be easily modified. Beyond design purposes, this kind of flexibility can be used at operations time to improve HSI in a continuous manner.

5. Human-Centered Design is a clear way to achieve HSI

Distinguishing task and activity: Cognitive Function Analysis (CFA). We have seen that SE is a matter of SoS and can be modeled using the agent representation, supported by cognitive function analysis (Boy, 1998). A cognitive function is defined as an entity transforming a task into an activity (i.e., from something prescribed into something effective) and can be applied to people or software artifacts. Functions can be deliberately defined, but they are really understood at operations time. When defined a priori, they are directly associated with tasks. Cognitive functions can only be defined when an agent produces the related activity. This is why task analysis is important as a first resource to the definition of functions, but it should be complemented by an associated activity analysis. This is not possible without simulation-based or real-world operations.

Comparison of the task and activity reveals the variability. Variability should be analyzed using the TOP model to determine if variability is due to technology, organization or people. Tasks are typically described using hierarchical decompositions. Activity is more difficult to describe because we usually capture events and actions traces that need to be interpreted and categorized into cognitive functions.

CFA consists in incremental construction and refinement of roles, contexts of validity and resources. Resources can be physical or cognitive functions. CFA is based on scenarios (stories) that represent the “real world” in which the tasks are to be performed as well as simulated or real-world traces of activities of the various agents involved. CFA is typically carried out on possible scenarios (what ifs) that make sense to be tested. It departs from classical causal analyses that are purely based on the continuity of the past. Prediction is typically short term because it is a derivative extrapolation (i.e., we cannot predict too far in time). It is event-driven. CFA is fundamentally goal-driven and creative. CFA contributes to create, design and refine cognitive functions based on experience and exploration of possible futures. Consequently, CFA can support HCD.

Departing from short-term prediction to testing possible futures. Risk taking is an issue in industry and government projects. We seek to decrease risks or at least understand them as much as possible. Predictive models have been proposed and used, but the predictions of today's technologies came from the creativity of science fiction writers (*cf.* How William Shatner Changed the World, 2005). Jules Verne (1873) wrote "Around the World in Eighty Days" more than a century before we were capable of going around the world in 90 minutes in the International Space Station. A focus on creativity and methods that enable the test of possible futures is necessary and preferable to trying to predict in the short term.

However, creativity alone will not achieve our engineering aims. Rather creativity must be melded with engineering practice to succeed. HCD is based on cognitive engineering, complexity analysis, organization design and management, human-computer interaction although we have focused on modeling and simulation (M&S) in this paper. HCD provides a methodology with depth and breadth to support development of design throughout the SE process.

In other words, conceptual thinking must be based on virtual prototypes, usability and usefulness tests on advanced simulators, to define appropriate requirements. This step is mandatory if we want to understand the complexity of HSI. It may be considered as more costly than what is done today in requirements engineering, but it will certainly be much less costly for the life cycle of a product. The current short-term, linear view of SE should be transformed into a longer-term nonlinear view incorporating HSI, with testing done on both virtual prototypes and real mockups, considering issues, capabilities and constraints of technology, organizations and people simultaneously.

Promoting modeling and simulation. Traditionally, engineering focused on designing and developing machines that were further tested by human factors and ergonomics specialists. Incorporating HSI into SE resulted in identifying and incorporating human capabilities and limitation in system development. HCD brings a methodology and philosophy to leading the design process prior to engineering and feeding forward solutions that can be successfully introduced into the society and organizations that will use the system. Requirements should be human-centered, but the main asset of modern technology is to provide M&S from the very beginning of a design project. We now have the tools to do so.

The software evolution that was initially seen as a difficult problem is now a resource for HCD. Human-centered designers can create, shape, refine and test new possible systems in a virtual world, and deduce appropriate requirements for the later manufacturing stages of the life cycle of a product. Dassault's Falcon 7X was entirely modeled and simulated using software for example. It can also be used during the whole life cycle to support risky decisions and operations while reducing risk. In the Rail domain, the Cab Technology Integration Lab (CTIL) has developed to test not only new integrations of controls, displays, and decision aids but also to explore new automation technologies and their HSI impacts to SE purpose (Jones et al., 2010).

6. Conclusion

Software is great at solving problems that are well stated, but cannot state (or solve) the messy problems people can. Therefore, the real issue of stating problems and imperatively taking abduction into account requires the art and technique of forecasting, projecting possible futures and evaluating their validity. This goal-driven approach is a fundamental alternative to the current event-driven short-term approach. Solutions are necessarily context-dependent and should emerge from a concurrent consideration of the TOP model.

Function allocation, using the strengths of each component is key and more productive than designing a system, which relies on the human flexibility to overcome flaws in architecture or design. We need to better understand human and systems roles, contexts of validity of these roles, and appropriate resources they can use. Interconnectivity involves numbers of nodes and links between these nodes. Complexity science tells us that behavioral attractors emerge from interactivity between these nodes therefore we need to discover these attractors and their emerging properties. Using HCD, risks can be mitigated by using M&S from the very beginning of design projects, and continuously during the life cycle of a technology. HCD introduces creativity and design thinking that SE urgently needs.

This paper promotes unification of HCD and SE towards better HSI. We need to acknowledge that engineering requires creativity as well as solid technical knowledge and skills. If we want to create a livable future, HCD will need to come first, anticipating engineering practice. Even engineering as it is thought today will evolve. Having good M&S, people will be able to develop systems because they will have support for both their functional and structural parts.

Money spent wisely during the early stages of a product minimizes technical, cost, and schedule risks, minimizing continuous costly repairs caused by design flaws once it is delivered. There will always be a tradeoff between procedural linear SE leading to rigid practices, and creative non-linear HCD providing flexibility and motivation. We propose an integration of HCD and SE where socio-technical leadership should return at the top of organizations, irrigating motivation and collective pride.

References

- Boy, G.A. (1998). *Cognitive Function Analysis*. Greenwood/Ablex, CT, USA; ISBN 9781567503777.
- Boy, G.A. (2005). Knowledge management for product maturity. *Proceedings of the International Conference on Knowledge Capture (K-Cap'05)*. Banff, Canada. October. ACM Digital Library.

- Boy, G.A. (2009). The Orchestra: A Conceptual Model for Function Allocation and Scenario-based Engineering in Multi-Agent Safety-Critical Systems. *Proceedings of the European Conference on Cognitive Ergonomics*, Otaniemi, Helsinki area, Finland.
- Boy, G.A. & Brachet, G. (2010). *Risk Taking*. Dossier of the Air and Space Academy, Toulouse, France, ISBN 2-913331-47-5.
- Boy, G.A. (2013). *Orchestrating Human-Centered Design*. Springer, U.K. ISBN 978-1-4471-4338-3.
- Brown, B. (2011). *Model-based systems engineering: Revolution or evolution?* Thought Leadership White Paper. IBM Rational. December.
- Clark, J.O. (2009). System of Systems Engineering and Family of Systems Engineering from a Standards Perspective. 3rd Annual IEEE International Conference on Systems Engineering. Vancouver, BC, Canada. ISBN 978-1-4244-3462-6.
- Fitts, P.M. (1951). *Human engineering for an effective air navigation and traffic control system*. Columbus, OH: Ohio State University Foundation Report.
- Harris, S.D., Ballard, L., Girard, R. & Gluckman, J. (1993). Sensor fusion and situation assessment: Future F/A-18 in A.H. Levis and I.S. Levis (Eds) *Science of Command and Control, Part III Coping with Change*. Fairfax, VA: AFCEA International Press.
- Haskins, C. (2011). International Council on Systems Engineering (INCOSE) Systems Engineering Handbook v. 3.2.2. A Guide for System Life Cycle Processes and Activities. INCOSE-TP-2003-002-03.2.2.
- Jones, M., Jones M., Olthoff, T. & Harris, S. (2010). The cab technology integration lab: A locomotive simulator for human factors research. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, September, Vol 54, No. 24, pp. 2110-2114.
- Maturana, H. & Varela, F. (1980). Autopoiesis and cognition: the realization of living. R.S. Cohen & M.W. Wartofsky (Eds.), *Boston Studies in the Philosophy of Science*, 42. Dordrecht: D. Reidel Publishing Co., ISBN 90-277-1016-3.
- Minsky, M. (1985). *Society of Mind*. Simon and Schuster, New York, NY.
- Mitchell, M. (2008). *Complexity: A Guided Tour*. Oxford University Press, New York.
- Paulk, M.C., Weber, C.V, Curtis, B. & Chrissis, M.B. (1995). *The Capability Maturity Model: Guidelines for Improving the Software Process*. SEI series in software engineering. Reading, Mass.: Addison-Wesley. ISBN 0-201-54664-7.
- Peirce, C.S. (1958). *Science and philosophy: collected papers of Charles S. Peirce, V. 7*, Harvard University Press, Cambridge, MA.
- Shatner, W. (2005). *How William Shatner Changed the World*. Directed by Julian Jones. Montreal: Handel Productions.
- Stacey, R. (1996). *Complexity and creativity in organization*. Berrett-Koehler, San Francisco.
- Verne, J. (1873). *Around the World in Eighty Days*, Série *Les Voyages extraordinaires*, J. Hetzel, Paris.