

What do we mean by Human-Centered Design of Life-Critical Systems?

Guy A. Boy

*Human-Centered Design Institute, Florida Institute of Technology
150 West University Boulevard, Melbourne, Florida 32901, U.S.A.*

Abstract. Human-centered design is not a new approach to design. Aerospace is a good example of a life-critical systems domain where participatory design was fully integrated, involving experimental test pilots and design engineers as well as many other actors of the aerospace engineering community. This paper provides six topics that are currently part of the requirements of the Ph.D. Program in Human-Centered Design of the Florida Institute of Technology (FIT.) This Human-Centered Design program offers principles, methods and tools that support human-centered sustainable products such as mission or process control environments, cockpits and hospital operating rooms. It supports education and training of design thinkers who are natural leaders, and understand complex relationships among **technology, organizations and people**. We all need to understand what we want to do with technology, how we should organize ourselves to a better life and finally find out whom we are and have become. Human-centered design is being developed for all these reasons and issues.

Keywords: Human-centered design, life-critical systems, cognitive engineering, complexity analysis, organization design and management, modeling and simulation.

1. Introduction

Who are we becoming in our highly-interactive Internet-supported world? We still formally belong to the “Homo Sapiens” species, which in Latin means “knowing human” or “wise human”¹. Are we becoming more knowledgeable and wiser? Modern information technology strongly supports knowledge development and learning. The Internet is a fascinating tool that enables people to learn from each other both in space and time. Connectivity is there directly with both living people and information previously stored. Information access is almost immediate, and people are able to interact with living and stored knowledge. Then, why in these conditions, are we not wiser than before? Perhaps we are but what does it mean to be wise today? People have built tools for a long time to support themselves in many kinds of activities. For the last fifty years, we have developed more tools than in the history of mankind. Also, technology development is accelerating. Are we mutating from “Human Beings” to “Technological Beings”? Jean

Pinet and I discussed this issue in the aeronautical context (Boy & Pinet, 2008.) Technology enables us to develop new kinds of skills such as flying. From a cognitive psychology viewpoint, humans are now augmented creatures that have new kinds of “cognitive functions” emerging from the use of technology. Cognitive functions can be represented by their role, context of validity and the resources that they need to be operated (Boy, 1998, 2011.) Aircraft technology led to the development of the aviation industry; new types of organizations emerged. Since we are now able to travel almost anywhere, anytime and much faster than before, our planet has become a global village. Consequently, new kinds of interaction among people have to be developed; we are still experiencing the very beginning of this, and many adjustments will need to be determined. This recent evolution is still under way, and it will require some time to get to a stable state.

This paper will present an approach to human-centered design (HCD) of life-critical systems, which is currently developed and taught at FIT in the HCD Ph.D. program. This paper will develop six major topics. (1) HCD is based strongly on an expansion of

¹ <http://en.wikipedia.org/wiki/Human>

cognitive engineering from single-agent models of cognition to multi-agent models of socio-cognitive interactions. Human and social sciences are needed to support understanding of HCD. (2) Life-critical systems (LCSs) include a large number of systems where people and machines interact with respect to three main principles: safety, efficiency and comfort. A comparison of various LCS domains is needed to better grasp these principles. (3) Contemporary human-centered design of life-critical systems (HCD-LCSs) cannot dissociate from the evolution and constant development of interactive media, e.g., the Internet. Computer science and human-computer interaction are needed to guide design and development choices. (4) LCSs are almost always complex with a larger number of components, highly non-linear and more interaction among these components. This is why mastering complexity analysis is so important. Complexity science, including mathematics and biology, is needed to better understand the nature of LCS complexity. (5) Since technology usage induces the emergence of new organizations, organization design and management need a human-centered approach also. Organization sciences and political science are also a must. (6) Finally, modeling and simulation (M&S) is extensive use to both foster creativity and rationalize LCS concepts. M&S also should be mastered and used during the whole life cycle of an LCS, and especially in the specifications of the requirements of both technology and organization.

2. Cognitive Engineering

Cognitive engineering is a composite discipline. Cognitive engineering deals with engineering and computer science. Additionally, it encompasses human and social sciences. Specifically, cognitive engineering is typically served well by disciplines such as cognitive science, artificial intelligence, human-computer interaction, human factors and ergonomics, as well as anthropology and organization sciences. Cognitive engineering takes human capabilities and limitations into account in systems' design. Such capabilities and limitation are typically expressed in terms of workload, fatigue, human errors, performance, risk taking, decision-making, situation awareness, and teamwork. Cognitive engineering became progressively an inescapable discipline mainly because computers introduced and established dominant cognitive work, as opposed to physical work. In addition, we needed to have workable methods to explain why trained and expert human oper-

ators can commit coarse errors. An important goal of cognitive engineering is to get socio-technical systems more reliable and resilient through cognitive modeling.

Cognitive science and psychology need models to support their studies and explain their experimental results. Cognitive science and psychology should take human factors into account during the whole life cycle of a product and mainly during the first stages of the design. We cannot analytically anticipate emerging cognitive functions during design, by definition! We then need to run human-in-the-loop simulations to discover and identify such functions or properties. These functions or properties emerge from interactions among the various agents. If these interactions are not effective, involved personnel will not figure out what these interactions are. The very concept of interaction promotes the multi-agent approach to socio-cognition. The big shift in cognitive engineering is right here today, i.e., from single-agent to multi-agent cognitive modeling.

In addition, designers should be aware of emergence. In complex systems, i.e., systems with numerous highly interconnected components or agents (often called systems of systems), interactions among components eventually generate emergent behavior and properties. First, designers need to understand the emerging cognitive function (ECF) concept, i.e., functions that components and agents need to have in order to achieve individual and collective goals. Second, designers need to understand that without appropriate **scenarios and criteria**, they will not recognize ECFs. Third, they need to be proactive and **take reasonable risks** to provoke the emergence of such functions. Design teams should be multidisciplinary, innovative and curious. Fourth, designers and engineers should train in a **human-centered perspective**. Designing the right function from the beginning is the key. Consequently, designers' cognition is a real research topic that needs more investigation. The term "human-centered design" is better than "user-centered design", because there are many people, i.e., humans, dealing with products for design, development, use, maintenance and eventually decommission. End-users constitute one of these categories of people.

3. Life-Critical Systems

A life-critical system (LCS) may have one, two or three of the following major dimensions, i.e., **safety**,

efficiency and **comfort**. In addition, we live in a world where **software** is everywhere. Software is used in cell phones, cars, public places, homes, and appliances of any kind including computer-aided design (CAD) tools. Software systems are not only used as tools; Software systems interact with people. Human-computer interaction (HCI) emerged as a field in its own right during the last three decades.

Comparing various kinds of life-critical systems is a good way to improve the design of life-critical systems. Metrics typically support the categorization of **emerging properties** of such systems. Appropriate organizational set-ups need to be put in place. Technology is critical. Legal and regulatory requirements need to be addressed and developed, for certification in particular. For example, life-critical avionics (i.e., electronic systems on aircraft) is produced using procedures based on standards that are incrementally upgraded. LCSs can be very simple systems such as circuit breakers, fire alarms, or a fuse. They also can be very sophisticated such as aircraft, spacecraft, nuclear power plants (NPP), operating rooms in hospitals, hospitals themselves and disaster management centers. Robotic surgery machines, NPP instrumentation and control (I&C), scuba-diving equipment, railway signaling and control systems, advanced braking systems on cars, air traffic management systems and space launch vehicle safety are life-critical systems.

These systems are characterized by several properties. They must be reliable, (human error and system failure tolerant and resistant) resilient, redundant, socio-cognitively stable, as well as controllable and observable. They require operator's training. They also induce the generation of appropriate operating procedures, specific human skills, human operator selection, regulations and heavy (more standardized) design and development processes.

This brings us to the difficult problem of **expertise**. Many contributions have been produced since the glorious time of expert systems and knowledge engineering, typically started during the early eighties (LaFrance, 1987; Boose & Gaines, 1988; 1990; Bradshaw, Ford, Adams-Webber, & Boose, 1993; Hoffman, Shadbolt, Burton & Klein, 1995). What do these contributions bring to the field of human-centered design (HCD)? Even if these contributions, situated at the intersection of artificial intelligence, cognitive psychology and philosophy, did not address HCD directly, they introduced a very valuable framework. Other contributions come from the HCI field (Sutcliffe, 1997), anthropology (Hutchins, 1995), and human engineering (Vicente, 1999).

All these theories are great, but they will never eliminate the involvement of domain experts in the design process. You cannot design an aircraft if you do not have both aerospace engineering and operational aviation practice knowledge and knowhow. We are back to the complexity chapter where complex things have to be handled by expert people. The cockpit in Figure 1 looks horribly complex; it is the Concorde cockpit with about 600 instruments for three crewmen. Guess what? Twenty Concorde aircraft were built (more concretely, 14 airline aircraft), pilots learned how to use them and did not have any accident with it in 27 years of operations, except the Paris accident in 2000 that was caused by an external unpredictable mechanical event. Concorde pilots were experts and they were able to manage the complexity of this aircraft. We need to mention that to become a Concorde pilot, they had to go through the whole chain of aircraft first, i.e. knowledge gained by experience was also considered as a requirement. Analogously, climbing Mount Everest requires experience climbing many other smaller mountains.



Figure 1. Concorde cockpit.

Complexity, expertise and life-critical systems are intimately related. I am regularly told by some of my friendly visitors that they do not understand how I can manage with what appears to be my cluttered desk, with many layers of documents, reports, books and post-its all over the place. It looks like a mess from the outside, but believe me I know where things are. Why? This is a matter of implicit **cognitive indexing**. Sometimes, documents are placed on top of each other chronologically. Other times, they are organized by topics or proximity of my current task. There are various kinds of organizations that look invisible, and therefore not understandable to outsiders, but are extremely effective in everyday life. In reality, the outsider's perceived complexity of my desk turns out to be a very organized internal com-

plexity in my brain... well, nobody can access my thoughts but me! I must admit that I get lost from time to time, just because I have overwhelmed my own cognitive indexing capacity. Also, this is why I use post-its to upgrade this implicit indexing of the documents on my desk to an explicit one. This is a concrete example showing that the way we observe the world is highly cognitive, depending on our own knowledge, knowhow and experience. This is why it is so important to construct cognitive representations and models that help us observe and manage life-critical systems in a more educated way.

In addition, I noticed that by increasing the size of my desk I did not reduced its perceived complexity. Indeed, I observed that I put even more documents on it. The desktop is a **cognitive attractor** (Lahlou, 2000). This concept is also valid for the metaphor of the desktop on our contemporary computers. It is so easy to put files on the desktop that we usually do it. The entropy keeps increasing until it is not acceptable any longer, and we have to clean it.

4. Advanced interaction media (AIM)

User interfaces are recognized as approximately 50% of work for real programs. User interfaces have a very important impact on the way products are made and used. User interfaces may contribute to the reputation of the organization that produces them. Be aware of any user interfaces safety issues. Usability engineering (Nielsen, 1993) has become a mandatory discipline in the production of software systems. According to the International Standard Organization (ISO), usability is “the effectiveness, efficiency, and satisfaction with which specified users achieve specified goals in particular environments” (ISO 9241-11.) This being said, be careful of the interpretation of this definition. Indeed, “users” are either novices, occasional users or experts in a given domain.

The HCI community grew from the early eighties with the use of the mouse, graphical displays, windowing systems and the developments of various kinds of widgets. The combination of hypertext and networked computer systems generated the World Wide Web. The evolution of such tools transformed our lives. Our long-term memory is now extended to the point that we are able to access almost any kind of information in a very short period of time. At the same time, we are able to access almost anybody on the planet easily and comfortably.

The shift from paper to electronic information is now almost definitive. Ubiquitous computing and cloud technology enable us to access our own information anywhere without carrying heavy hardware; handheld devices are sufficient. Advanced interaction media enable mobility and flexibility. In addition, information can be accessed in context and context-aware systems enable faster access to information. Radio Frequency Identification (RFID) has become a very useful indexing technology that enables new types of automation. Specifically, RFID increases the potential for context-sensitive information systems.

Advanced interaction media (AIM) are computer-based systems that enable interaction with information and knowledge, in either a vivid form coming from people directly or various kinds of data formats. Computer-supported cooperative work research contributed to the development of AIM devoted to human-human communication, not to mention that constantly increasing amounts of data require new kinds of visualization techniques and tools.

Specifically in return, human-centered design benefits from advanced interaction media support. Developing and testing a mockup contribute to increased usability from the early stages of design. Active design documents enable storage of design history and support traceability during the later stages of the development of a product (Boy, 2005). AIM facilitates product integration by simulation, and even facilitates product management (Coze et al., 2009). In many cases, AIM is the most important part of the products such as the iPhone or the iPad.

Today, software-based interactivity is almost everywhere. New cockpits are qualified as “interactive cockpits”, not because pilots interact directly with mechanical parts of the aircraft, but because pilots interact with a pointing device enabling the management of computers, which interact with mechanical parts such as rudder, flaps and slats. AIM therefore mediates interactions between people and their surrounding physical world. Consequently, it is crucial to better understand the nature of AIM to overcome increasing lack of situation awareness and decision-making problems. The next section will describe why and how to analyze complexity of LCSs, where AIM are even more integrated today.

5. Complexity analysis

Complex systems exist in the natural world, and scientists have studied such complexity for a long

time, trying to figure out persistent behavioral patterns, identify generic phenomena, relationships among components and so on. Today, technological developments have brought new kinds of complexity. In other words, scientists have developed **models** in order to “simplify” complexity. For example, a long time ago people used to watch the sky as a complex set of stars; Copernicus, Kepler and Galileo simplified complexity by providing models based on two simple variables, i.e., distance and mass. In the same way, scientists develop models in order to describe what they observe, i.e., they rationalize complex datasets.

Complexity is often a matter of knowledge or **ignorance**. The more we know about something, the more we find it simple to understand and use. For example, you may have a hard time driving to a place for the first time. You may need a map and directions. Specifically, you may need to think, concentrate and act appropriately. Once you have driven to this place, it becomes easier. You constructed a pattern (or model) of the way to get to this place. You may refine this way by optimizing time and distance. In fact, complexity was mainly in your ignorance of the salient entities and the relationships among them, which determined the relevant pattern. Discovering and/or learning patterns/models tend to deconstruct complexity.

We actually simplified flying when we understood that flying was a matter of thrust and lift. Once we had the right structural devices to insure the lift, and the right propulsion to insure the thrust, we managed to fly. Clement Ader was one of the first to make this point. He built the first flying machine *Éole* that he attempted to fly on October 9, 1890 in the vicinity of Paris. He flew 50 meters reaching a height of 20 centimeters. This was 13 years before the Wright Brothers (AAE, 1994). At this time, Ader deconstructed the complexity of human flight by proving that it was possible to physically stay in the air propelled by an engine. After Ader’s work, many other people designed, developed and used aircraft along these lines, and refined the manned-flight concept, i.e., they mastered the complexity of the manned-flight concept both functionally and structurally. Note that we incrementally learned from experience. Also, it is important to mention that aircraft improved over the years from safety, efficiency and comfort points of views because they were developed symbiotically with pilots. More importantly, the experimental test pilot (ETP) job emerged from the human-centered design approach, even if it was not formalized as it is today.

HCD is about artifact design and use. **Artifacts**, i.e., artificial things, differ from natural things because people build them. They can be simple, but this paper focuses on complex artifacts, and more precisely complex systems. Systems are usually defined as an integrated set of interconnected components. Examples of systems may include aircraft, spacecraft, nuclear power plants, medical operating rooms and disaster-management centers. These systems are inherently complex because they are cognitive prostheses (Ford, Glymour & Hayes, 1997). An aircraft is a cognitive prosthesis because it is designed to supply people with flying capacity (remember that people do not fly naturally!)

I will make the distinction between intrinsic complexity and extrinsic complexity. **Intrinsic complexity** results from system architecture and internal relationships among its components. **Extrinsic complexity** results from the activity of the system, i.e., interaction between the system and its environment including its users. Of course, intrinsic complexity and extrinsic complexity are closely related. First, we will try to analyze them separately. Next, we will show how they relate to each other.

Technology is supposed to make our lives safer, more efficient and comfortable, but in reality we observe that technology creates new kinds of security and safety issues, catastrophic events and complicated ways of living. I am not an anti-technology activist! My research work focuses on HCD where it is important to involve the right people in the design of complex life-critical systems. This humanistic approach to design and engineering forces the development of new integrative methods, techniques and tools that encapsulate the **Technology, Organization and People (TOP)** approach.

For a long time, systems were designed individually without much attention to their coupling to other systems and most importantly people. Their integration into an organizational environment was always the source of various kinds of surprises and discoveries, with a few catastrophes. Adjustments were always necessary and often systems had to be redesigned to fit the reality of their actual use. This process seems natural and acceptable during development phases, but this variance is not acceptable when the system is delivered. This is a question of maturity. Maturity is **technology maturity** and **maturity of practice**. The former deals with intrinsic complexity; the latter deals with extrinsic complexity.

In design and engineering, maturity is strongly related to the nature of high-level requirements. When high-level requirements are “well-defined”, it is like-

ly that the resulting system is good, with respect to appropriate principles and criteria. However, when high-level requirements are “badly-defined”, it is, in most cases, that the resulting system is not acceptable. In addition, the nature and the number of tests performed before delivery are crucial to insure the maturity of the system. High-level requirements are not defined well without a human-centered design approach that enables the designer to create a holistic view and actual use of the targeted product. Modeling and simulation are key methods, techniques and tools to achieve this goal. They enable an **outside-in approach** to design, instead of the classical **inside-out approach** of current engineering practices, where systems design is technology-centered, seldom taking into account user experience and ergonomic design factors. The inside-out approach created the need for user interface approaches where such an interface attempts to compensate and ultimately hide intrinsic complexity, and sometimes force users to adapt to the system. Conversely, the outside-in approach takes extrinsic complexity into account from the early stages of the design process.

We have used the inside-out approach for a long time because early engineering systems were simpler than systems that we know today. Therefore, it was easy to design a user interface because the number of variables and parameters was reasonably small. Today, this number has become huge, and designers have to make difficult choices in the design of user interfaces. What is important to show? What should be controlled? Layers and layers of software are developed to take into account safety, efficiency and comfort, while at the same time increasing both intrinsic and extrinsic complexity. Consequently, systems have their own behaviors that people need to perceive, understand, consider and react to appropriately. We have moved into a human-system interactive world not considered as a single human operator using a machine, but as a multi-agent interactive environment.

In the design of a complex system, we have to take into account not only its intrinsic complexity but also and foremost its extrinsic complexity. Without designing the overall architecture and enabling its functionalities and behaviors, it is not possible to access its extrinsic complexity (nor its intrinsic complexity also!) Therefore, instead of designing systems individually and then integrating them into an environment, it is better to adopt the **systems-of-systems (SoS)** approach. The SoS concept originated in the defense sector (Luzeau & Ruault, 2008.) Several different definitions are given by many authors (Jamshidi,

2005), but these definitions require more research to arrive at a workable definition. Today, we cannot think of the air traffic management of the future without stating it in terms of SoS. Other examples are the Internet, intelligent transport systems and enterprise information networks (or Intranets). The SoS concept now integrates the distinction of intrinsic and extrinsic complexity.

Since SoSs consist of humans and systems, original natural systems tend to become artificial systems. Many examples have emerged from this evolution such as genetically modified organisms and integrated prostheses. Technology is now part of our lives and needs to be investigated appropriately if we want to take care of humanity and the Earth. Improving our understanding of complexity is very important when we, humans, attempt to modify natural complexity. Consequently, **life-critical systems** need better investigation and understanding in order to find the right mix between humans and machines.

In fact, the classical **positivist approach** is no longer sufficient, and often ineffective and inappropriate, and we prefer to use more **phenomenological approaches** to design and development. HCD requires us to emphasize the positivism-phenomenology distinction². In particular, a complex system is an articulated set of dedicated sub-systems that induces emerging phenomena that each of these sub-systems does not have initially. Good human-centered design should focus on the discovery of these emerging phenomena.

6. Organization design and management

In this section, we will uncover the difficult problem of organization influence on human-centered design. Many authors already investigated and analyzed such problem. In the human error arena, Rea-

² The French philosopher and sociologist Auguste Comte introduced positivism in the beginning of the nineteenth century. Positivism asserts that the only authentic knowledge is that which is based on sense experience and positive verification. The German philosopher Edmond Husserl introduced phenomenology in the beginning of the twentieth century as the study of consciousness and conscious experience. Among the most important processes studied by phenomenology are intentionality, intuition, evidence, empathy, and intersubjectivity. The positivism-phenomenology distinction opens the debate on objectivity and subjectivity. Our occidental world based most of our design and engineering on positivism which led to developing a very precise and verifiable syntax, often leaving semantics somewhere behind, may be because semantics is full of subjectivity. It is time to re-qualify phenomenology in design and engineering. A few organizations and companies already work in this direction.

son (1990) provided the famous Swiss cheese model. This model provides a macroscopic view of what it requires to anticipate potential accidents by developing appropriate defenses. However, this model does not tell us how the organization should be designed to manage socio-technical reliability. Kletz (2000) addressed the influence of senior management on functional risk, and its involvement in the detail of safety issues with their close attention to output, costs and even product quality.

Vaughan (1996) examined the origins of the 1986 Challenger space shuttle accident. She pointed out that the accident was caused mainly by social structures: “a mistake embedded in the banality of organizational life.” **Organizational routine and complacency** lead to accidents. Specifically, Vaughan showed how “deviance in organizations is transformed into acceptable behavior.” We are talking about the culture of the organization itself, and the role of the senior management that needs to ensure this safety culture glue among the actors involved. Of course, an accident does not happen, usually, from only one cause, but the main issues remain senior management’s situation awareness and appropriate decisions.

Use statistics carefully. NASA management estimated safety as 10^{-5} accident per space shuttle launch, and in contrast NASA engineers estimated safety as between 0.5×10^{-2} and 10^{-3} (Feynman, 1989). In reality, it has been 2 on 135 launches, i.e., around 2×10^{-2} per launch. Of course, organizational decision-making was carried out as though management’s estimate was correct. This brings to the front the issue of **management authority and competence**. Sometimes, the distance between the top management and the real technical competence can be large. This is why it is crucial to make sure that technical competence is present at the highest level of the organization.

HCD of a product is not developed properly if the organizational aspects are not taken into account in both the structure where design and development are performed, and the organization that will receive the product designed. In the first case, we will talk about **organizational genotypes** of the product; in the second case, we will talk about **organizational phenotypes** of the product (Figure 2.) Organizational genotypes include the way the organization influences product design, development, work assignment, articulation between technical and financial management, planning, quality control, evaluation, certification, external relations and personnel training (this is a non exhaustive list of course.) Organiza-

tional phenotypes include the way the recipient organization integrates, operates and maintains the product.

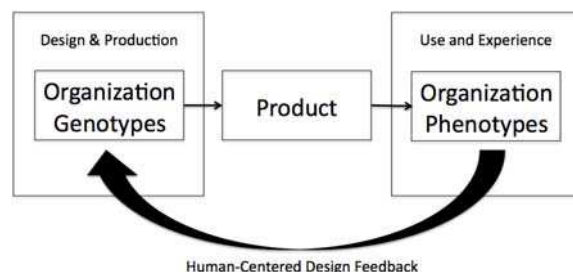


Figure 2. Organization genotypes and phenotypes of a product, and human-centered design feedback.

System design and development is a matter of networked component production and integration. The organization producing the system is defined as an **autopoietic system**, in Maturana and Varela’s sense (Varela, Maturana & Uribe, 1974; Maturana & Varela, 1987.) This system produces a system that includes its own genes. An autopoietic system is producer and product at the same time. Following the autopoietic model, human-centered design is both the process of design and the design itself, i.e., a human-centered system design is incrementally modified in a participatory way by the various actors or agents who deal with it. HCD feedback is not only a matter of fixing the values of appropriate parameters; it contributes to regenerate the product itself. This biological definition of the genesis of a system encapsulates organizational genotypes and phenotypes. The HCD feedback takes into account emergent organization phenotypes in the redesign of organization genotypes and therefore the product itself.

Now, technology is almost always equipped with layers of software that enable machines to interact with humans, at least in very limited contexts. We commonly talk about human and machine agents (HMA). Human-computer interaction was traditionally thought of as a person facing a computer in a one-to-one relation. Today, there are HMA networked societies in the sense of Minsky (1986), i.e., an agent being a society of agents. Human modeling, often commonly thought of as information processing (Newell & Simon, 1972; Wickens, 1992,) progressively migrated towards multi-agent organization modeling where cognition is distributed (Hutchins, 1995.)

In addition, the evolution of practices due to automation within an organization required developing a model that could support human-centered design of

multi-agent systems. The cognitive function concept (Boy, 1998) emerged as a useful representation to support such modeling of both individuals (i.e., individual agents as organized structures of cognitive functions) and organizations (i.e., a set of cognitive functions distributed among a set of agents). Examples of cognitive functions are “speed control,” “trajectory management,” “collision avoidance” and “failure detection and recovery.” Function allocation in life-critical (human and machine) systems is not a new research topic (Fitts, 1951.) However, there is a need for unifying what is separately done in engineering, human factors and organization science.

The Orchestra model (Boy, 2009) suits well this kind of evolution where (1) agents require a common frame of reference (a music theory analog); (2) contracts (scores) must be appropriately and formally coordinated (the role of the composer); (3) real-time coordination must be assured (the role of the conductor); and (4) agents must have specific abilities to perform according to contracts (role and proficiency of the musicians.) Contracts are seen as scenarios or storyboards, with an additional responsibility dimension. Consequently, the Orchestra model will support the identification of both deliberate and emergent cognitive functions during the life cycle of a multi-agent life-critical system. It is based on previous work on cognitive function analysis (Boy, 1998, 2011) and function allocation work (Grote et al., 2000.)

This metaphoric model was motivated by the introduction of new concepts of operations (CONOPS) in air traffic management (ATM), such as task delegation from ground controllers to flight crews. This kind of transfer of authority is typically expected to induce the emergence of new cognitive functions among the various ATM agents whether they are humans or automation. More specifically, the current hierarchical model of air traffic control (ATC), where authority is centralized on the ground, is evolving toward a distributed model of authorities that need to coordinate among agents that are evolving also. More generally, authority has become the central driving force of the Orchestra model, where authority encapsulates both control and accountability.

7. Modeling and simulation (M&S)

Designing a system requires both creativity and rationalization. These two processes are antagonistic. The former is about divergent thinking, generation of

ideas and brainstorming processes when it is done by a group of people. The later is about convergent thinking, analysis of ideas and synthesis into concepts, evaluation and prioritization of concepts. Generating concepts requires formalizing them. This is where modeling enters into play. We need to have the right conceptual tools to share ideas and concepts.

What is a model? A **model** is a simplification of the reality, e.g., a system or an environment, which we try to represent. Therefore a model is a simplified representation that puts forward a few salient elements and their relevant interconnections. If the model takes care of the interconnections among system components, we need the **simulation** to take care of the interaction among these components. In other words, simulation brings the model to life.

Modeling and simulation (M&S), as a discipline, enables understanding of the purposeful components, their interconnections and interactions. This brings to mind the issue of the level of detail and granularity of the representation being used in chosen models. When you are developing a model, you are always facing a tradeoff in choosing the right level of **granularity**, i.e., what is meaningful against what is unnecessary. If you stay at too high a level, you might miss interesting interactions during the simulation process. Conversely, if you want to model every detail, you might run into very complicated interactions, which are extremely difficult to understand. The model of everything does not exist and will never exist. Stay focused on the **purpose!** There are various kinds of models, and we will see the main distinctions that need to be understood into order to use and/or combine them appropriately.

Simulation is typically computer-supported and generally, but not necessarily, interactive, e.g., it could be paper-based or part of a brainstorming session such as a role-playing game. As already discussed, simulation is used to improve understanding of the interactions among the various elements of the model that it implements and simulates. Simulation is used also to improve the model and eventually modify it (Figure 3). Simulators enable people to engage into the activity that the system modeled enables them to perform, e.g., driving a car simulator. You may use simulation to learn about a domain, such as flying. Typically, simulators are upgraded as we learn more about this domain. In addition, simulation enables human operators to “experience” situations that are not experienced in the real world such as possible accidents. As shown on Figure 3, modeling a system also takes into account people involved and other relevant systems interacting with each other.

The modeling process produces a model that can be simulated on a simulator, which in turn produces experimental data. These data are usually analyzed. Data analysis enables potential identification of emerging properties, which enable learning about system use. The M&S design cycle shows that modeling is a closed-loop process that in turn enables system re-design, modification of people practices/profiles and potential re-definition of other systems.

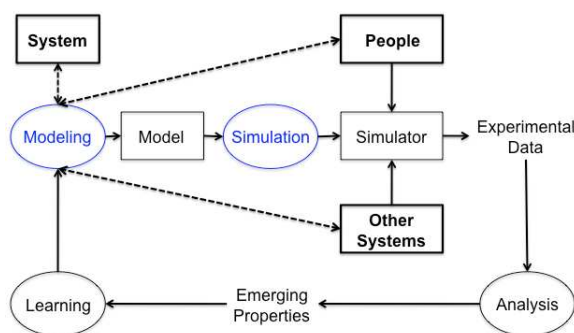


Figure 3. The M&S design cycle.

Simulation is **imitation**. The first time I observed the Concorde simulator at Aeroformation was in Toulouse, France, in the late seventies. The view was generated from a small camera moving over a giant model landscape that was fixed to the wall in a big hangar. The images from this camera were projected onto large screens in front of the cockpit windows. The original simulation setup was limited to a single airport. Then came computer-generated images that totally changed the possibilities of flight simulation. This kind of simulator was used only for training.

Also, simulation is used to explain and aid difficult decision-making processes. For example, modeling and simulation were used very much at the Kennedy Space Center to figure out launch pad configurations prior to a shuttle launch. Current M&S tools, such as CATIA and DELMIA, for example, enable the visualization of complex structures and functions. NASA engineers and management use resulting simulations as mediating representations and decision support tools.

Disney and Universal, for example, use simulation for a totally different purpose, i.e., entertainment. In addition to attempting to provide natural sensations and “real-world” experience, Disney and Universal create fiction. It is amazing how people can manage both simulated “natural” things and fictive objects brought to life through simulation. Computers and

software make this kind of mix possible, but there are techniques and art to make this kind of technology acceptable to people and provide them with a memorable experience.

There are several kinds of simulation that involve both real and simulated people and systems interacting among each other. The term human-in-the-loop simulation is broadly used in this case. The industry can simulate various kinds of effects such as visual scenes, sounds, motion, and odors. The videogame industry is progressing very fast to integrate these sensations.

8. Conclusion and perspectives

Human-centered design of life-critical systems is a discipline under development. The HCDi of FIT advances future generations of designers and leaders of life critical systems through outreach, education and research in science, engineering and human and social sciences. Our interdisciplinary approach promotes innovation and leadership to synthesize emerging properties of technology, organizations and people for safe and efficient system design. We provide a single point of contact for students, faculty, funding agencies and businesses.

HCDi members are faculty, permanent and visiting research scientists and graduate students who have strong backgrounds in complex systems (e.g., cockpit design, aircraft certification, air traffic control and management, space systems, space shuttle engineering) and science (e.g., physics, mathematics, psychology and education.) HCDi members conduct research in cognitive engineering, AIM, complexity analysis in human-centered design, LCSs, human-centered organization design and management and M&S.

Projects carried out at HCDi include: (1) Nuclear control room design, i.e., cognitive function analysis and allocation, human-centered automation, advanced interaction media, eye tracking experimentations; (2) The virtual camera in planetary exploration, i.e., multi-source data visualization techniques; (3) A multi-source weather awareness system for future cockpits that is expected to improve flight condition situational awareness by design; (4) Design of air traffic management systems and complexity analysis of Trajectory Based Operations for NextGen; and (5) Design of instructional technology for the online learning environment.

Human-centered design of life-critical systems will make the world a better place through the creation of engineering leadership, and a new motivating development of science, technology, engineering and mathematics (STEM). Apply human-centered design throughout the entire life cycle of LCSs. We also need to educate key leaders about HCD practices. Finally, we need to develop a broad community of HCD experts collaborating across academia and industry.

Acknowledgements

Many thanks to the lecturers and researchers who took the time to provide the necessary information to illustrate advanced interaction media, life-critical systems, complexity analysis and organization design and management in the scope of the HCD Ph.D. program at FIT: Thierry Bellet, Jeffrey Bradshaw, Kerry Gilger, Gudela Grote, Jonathan Grudin, John Hansman, Daniel Krob, Paul Krois, Ludovic Loine, Patrick Millot, Amy Pritchett, Antoine Rauzy, Myroslav Sparavalo, and Robert Wears. I also would like to thank Sharon Chinoy, Carol Craig, Rhonda Lyons, Donald Platt, Cynthia Schmitt, Kara Schmitt, Lucas Stephane, for their support and active involvement in HCD-LCS.

References

- [1] AAE. (1994). *Au temps de Clément Ader*, ouvrage coordonné par l'Académie de l'Air et de l'Espace. ISBN 2-87717-044-6.
- [2] Boose, J. H., & Gaines, B. R., (Eds.), (1988). *Knowledge Acquisition Tools for Expert Systems, Knowledge Based Systems Vol. 2*. San Diego, CA: Academic Press.
- [3] Boose, J. H., & Gaines, B. R., (Eds.), (1990). *The Foundations of Knowledge Acquisition, Knowledge Based Systems, Vol. 4*. San Diego, CA: Academic Press.
- [4] G.A. Boy (Ed.), (2011). *Handbook of Human-Machine Interaction: A Human-Centered Design Approach*. U.K: Ashgate,...
- [5] G.A. Boy (Ed.), (2010-2011) Six courses taught at Florida Institute of Technology, as requirements for the Human-Centered Design Ph.D. program
- [6] Boy, G.A., & Brachet, G., (2010). *Risk Taking*. Dossier of the Air and Space Academy. ISBN 2-913331-47-5, Toulouse, France: .
- [7] Boy, G.A., (30 September-2 October 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.
- [8] Boy, G.A., & Pinet J., (2008). *L'être technologique (The Technological Being)*. L'Harmattan, Paris: .
- [9] Boy, G.A., (October, 2005). Knowledge management for product maturity. Proceedings of the International Conference on Knowledge Capture (K-Cap'05). Banff, Canada.
- [10] Boy, G.A., (1998), *Cognitive Function Analysis*. Greenwood/Ablex, CT, USA,.
- [11] Bradshaw, J. M., Ford, K. M., Adams-Webber, J.R., & Boose, J. H. (1993). Beyond the repertory grid: New approaches to constructivist knowledge acquisition tool development. *International Journal of Intelligent Systems*, 8, pp. 287-333,
- [12] Coze, Y., Kawski, N., Kulka, T., Sire, P., Sottocasa, P. & Bloem, J., (2009). *Virtual Concept - Real Profit with Digital Manufacturing and Simulation*. Dassault Systèmes and Sogeti, ISBN 978 90 75414 25 7, .
- [13] Feynman, R.P., (1989). *What Do You Care What Other People Think?* Unwin Hyman, UK.
- [14] Fitts, P.M. (Ed.) (1951) *Human engineering for an effective air navigation and traffic control system*. National Research Council, Washington, D.C.
- [15] Ford, K. M., Glymour, C., & Hayes, P. J. (1997). Cognitive Prostheses. *AI Magazine*, Vol. 18 (3), Fall,
- [16] Hoffman, R. R., Shadbolt, N. R., Burton, A. M., & Klein, G. (1995). Eliciting knowledge from experts: A methodological analysis. *Organizational Behavior and Human Decision Processes*, 62, pp. 129-158,
- [17] Hutchins, E. (1995). How a Cockpit Remembers its Speeds. *Cognitive Science*, 19, pp. 265-288
- [18] Grote, G., Ryser, C., Wäfler, T., Windischer, A. & Weik, S., (2000). KOMPASS: A method for complementary function allocation in automated work systems. *International Journal of Human-Computer Studies*, 52, pp. 267-287,.
- [19] Jamshidi, M., (2005) *System-of-Systems Engineering - A Definition*. IEEE SMC, Oct., pp. 10-12., http://ieeesmc2005.unm.edu/SoSE_Defn.htm.
- [20] Kletz, T., (2000). *An Engineer's View of Human Error*. Third edition, Institution of Chemical Engineers, London: .
- [21] LaFrance, M., (1987). The knowledge acquisition grid: A method for training knowledge engineers. *International Journal of Man-Machine Studies*, 26, pp. 245-255.
- [22] Lahlou, S., (2000). *Attracteurs cognitifs et travail de bureau. Intellectica: revue de l'Association pour la Recherche Cognitive*, (30). pp. 75-113. ISSN 0769-4113.
- [23] Luzeau, D. & Ruault, J.R. Eds., (2008). *Systems of Systems* ISBN 978-1-84821-164-3. John Wiley & Sons, Inc., Hoboken, NJ, USA:.
- [24] Maturana, H. R. & Varela, F. J., (1987). *The tree of knowledge: The biological roots of human understanding*. Boston: Shambhala Publications.
- [25] Minsky, M., (1986) *The Society of Mind*. New York: Simon and Schuster,.
- [26] Nielsen, J., (1993) *Usability Engineering*, Boston, MA: Academic Press .
- [27] Newell, A. & Simon, H. A., (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall.
- [28] Reason, J., (1990). *Human error*. Cambridge University Press.
- [29] Sutcliffe, A. G., (1997) *Task-related information analysis*. *International Journal of Human-Computer Studies*, 47 (2), pp. 223-257,.
- [30] Varela, F., Maturana, H. & Uribe, R., (1974). Autopoiesis: The organization of living systems, its characterization and a model, *BioSystems*, Vol. 5), pp. 187-196.
- [31] Vaughan, D., (1996). *The Challenger Launch Decision*, Chicago: University of Chicago Press.
- [32] Vicente, K. H., (1999). *Cognitive Work Analysis: Towards safe, productive, and healthy computer-based work*. Mahwah, NJ: Lawrence Erlbaum Associates,.

[33] Wickens, C.D., (1992). *Engineering Psychology and Human Performance*, (2nd Edition), New York; Harper Collins,.