Annual Reviews in Control 38 (2014) 1–11

Contents lists available at ScienceDirect

Annual Reviews in Control

journal homepage: www.elsevier.com/locate/arcontrol

Review

From automation to tangible interactive objects

Guy André Boy

Human-Centered Design Institute, Florida Institute of Technology, 150 West University Boulevard, Melbourne, FL 32901, USA

ARTICLE INFO

Article history:
Received 16 October 2013
Accepted 12 February 2014
Available online 13 April 2014

ABSTRACT

Automation led to many innovations for a long time, most of them were developed during the twentieth century. It was commonly thought as a layer on top of a mechanical system. It promoted system management over low-level control. The more information technology evolves, the more it takes a fundamental part in our lives. This article describes a paradigm shift where automation will no longer be an add-on, and where software supports the definition, implementation and operationalization of functions and structures of products from the beginning of the design process. Any design today starts by using computer-aided design tools that enable us to easily draw, modify and fine-tune any kind of system. We can fully develop an airplane and literally fly it as a complex piece of software. Usability and usefulness can be tested before anything physical is built. Consequently, human-centered design (HCD) is now not only feasible but also can drive the overall engineering of products. We have started to design products from outside in, i.e., from usages and purposes to means. We even can 3D print mechanical parts from the software-designed parts with ease. In human–computer interaction, specific research efforts are carried out on tangible objects, which define this inverted view of automation. We now design and develop by using information technology to do mechanical things, and therefore redefine the essence of a new kind of cognitive mechanical engineering. This article is about the revolution that is currently happening in engineering and industrial design due to the immersive influence of computers in our everyday life, and the expansion of HCD.

Contents

E-mail address: gboy@fit.edu

http://dx.doi.org/10.1016/j.arcontrol.2014.03.001
1367-5788/

1. Introduction

This article is an extension of a keynote given during the IFAC Human–Machine Systems conference, held in Las Vegas on August 14, 2013. The title of the initial talk was: "Human Systems Integration: Unifying Systems Engineering (SE) and Human Centered Design (HCD)." Since the keynote explained the shift from automation (i.e., including information technology and control theory into mechanical systems) during the twentieth century, to tangible interactive objects (TIOs, i.e., providing physical shape to and grasp of software artifacts), it was decided to reshape the title of this article, which also follows up the conclusion of the Handbook of Human–Machine Interaction that emphasizes the shift from automation to interaction design (Boy, 2011). However, the unification of SE and HCD remains a major component of the essay.

A TIO is a robotic artifact, ranging from a piece of software to a physical artificial agent, which has reasoning and/or reactive computational features and, therefore a role, a context of validity and appropriate resources (i.e., cognitive functions, later defined in the article). We now find TIOs in various kinds of habitats, vehicles, public places and industry. TIOs are the result of the evolution of computer science and engineering toward ubiquitous and pervasive computing, where computers make themselves invisible buried into appliances and systems of any kind (Mark, 1999; Weiser, 1991). Recent development of *modeling and simulation* (M&S), high connectivity, 3D printing and TIOs enable effective human-centered design, leading to human-systems integration. Making a TIO is no longer automating a previously developed physical object or machine; it is progressively designed, from the start, as a software object that is transformed into a physical entity.

Automation led to many innovations for a long time, most of them were developed during the twentieth century. More than thirty years ago, advanced automation enabled the shift from three to two crewmen in commercial aircraft cockpits and led to the glass cockpit concept (Boy & Tessier, 1985). This article is based on this initial experience as well as on nearly all Airbus cockpit designs and evaluations from the A 300 FF (Garuda) to the A 380 (Boy, 1998a, 1998b, 2011). It is also based on automation experience in other domains such as US Space Shuttle and Space Station procedure following and documentation systems (Boy, 1987, 1991), the NASA Lunar Electric Rover (the LER was renamed Space Exploration Vehicle) design and more specifically its navigation system (the Virtual Camera project; Boy & Platt, 2013), various control rooms in nuclear, telecommunications and aviation industries (Boy, 2011), and most recently the design of an interactive rocket launch control room at Kennedy Space Center. This 35-year experience is the main ingredient for a vision of the shift from automation to TIOs, and analysis of the mutual influence of engineering, information technology, human and social sciences, and design.

The organization of this article is as follows. In Section 2, the evolution of automation is presented. The cognitive function model is described to support a better definition of automation reactions to expected and unexpected events, as well as function allocation and the concept of emergent cognitive functions. In Section 3, human-centered design is explained and reasons are given why it is now possible. In Section 4, it is shown how the V-model can be transformed to integrate HCD and SE. It is shown why the concept of user interface is a wrong concept when it is used at the end of a design and development project instead of starting by analyzing, designing and evaluating technology, organizations and people's jobs holistically from the beginning. In Section 5, the Orchestra organizational model supporting HCD is presented. It is based on a multi-agent approach and cognitive engineering principles. It is very important at this point to operationalize the cognitive function concept. Section 6 is devoted to discussions on the shift. In Section 8, some concluding remarks are given.

2. Evolution of automation

The Bing dictionary provides an interesting definition of *automation* that deals with the "replacement of human workers by technology: a system in which a workplace or process has been converted to one that replaces or minimizes human labor with mechanical or electronic equipment." Automation has several synonyms such as mechanization, computerization and robotics (http://www.bing.com). Automation has lots of advantages such as increasing productivity, quality, robustness, consistency and product returns (mainly by decreasing costs). Sheridan contributed to describe and foster the evolution of automation (Sheridan, 1992, 1997, 2002). Automation also has some issues such as rigidifying practices, increasing complacency of people involved in supervisory control, decreasing and sometime removing human skills (Billings, 1991). Let's further describe automation using a more formal approach supported by the cognitive function formalism.

2.1. Cognitive functions

Automation can be described as a transfer of *cognitive functions* from people to machines (Boy, 1998b). A cognitive function is defined by three attributes: *role*, *context* of validity and necessary *resources* supporting the use of it. A cognitive function enables the execution of a task and produces an activity. Therefore, the input of a cognitive function is a task, and its output is an activity. Using this definition, we can characterize the activity of a human or a system who/that has to execute a task (Fig. 1). This formalism was successfully used to help figure out the various roles that are transferred from people to systems, as well as in what contexts they are valid and what resources they need to perform a given task, e.g., in aircraft automation (Boy, 1998a).

Cognitive functions are very similar to Leont'ev's *functional organs* (Boy, 2002; Leont'ev, 1981). These concepts were developed by the Russian activity theory school, and by Alexei Leont'ev and Victor Kaptelinin in particular. "Functional organs are functionally integrated, goal-oriented configurations of internal and external resources. External tools support and complement natural human abilities in building up a more efficient system that can lead to higher accomplishments. For example, scissors elevate the human hand to an effective cutting organ, eyeglasses improve human vision, and notebooks enhance memory. The external tools integrated into functional organs are experienced as a property of the individual, while the same things not integrated into the structure of a functional organ (for example, during the early phases of learning how to use the tool) are conceived of as belonging to the outer world." (Kaptelinin, 1995).

Cognitive function analysis (CFA) is then a very useful approach and method to better understand how functions can be allocated among humans and systems. CFA was defined and developed

Fig. 1. A cognitive function defined as transforming a task into an activity, and being defined by a role, a context of validity and a set of resources.

during the nineties to support analysis, design and evaluation of highly automated cockpits (Boy, 1998a). It is now commonly used in the design of life-critical systems (Boy, 1998b, 2011, 2013; Boy & Ferro, 2003). CFA requires the development of two types of scenarios: declarative scenarios (i.e., possible configurations in terms of roles and resources), and procedural scenarios (i.e., possible chronologies or timelines in terms of contexts). A generic CFA typically starts by developing (1) an ontology of resources in terms of human and machine agents and relationships among them, and (2) a timeline of the various events of a generic scenario, and incrementally elicit relevant cognitive functions (Fig. 2).

However, CFA cannot be limited to a deliberative description of the functions initially allocated to people and machines. It must be used to guide, and analyze the results of, experiments leading to the elicitation of cognitive functions that emerge during operations. It is fundamental to make a distinction between deliberatively allocated functions and emerging cognitive functions. The later cannot be defined in the first place without experience of various interactions among human and artificial agents involved. In other words, we cannot say that we are doing human-centered automation without looking for emerging cognitive functions experimentally. CFA is then used to support a cognitive function repository that is incrementally upgraded.

2.2. Dealing with the unexpected

Automation was based on predictive causal approaches for a long time. It was dominated by Laplace functions, linear equations and Kalman filtering for example. These approaches are based on very precise mathematical models that only work on close-world systems in very specific short-term contexts. They worked very well at the skill-based behavioral level (Rasmussen, 1986), i.e., at the perception–action, stimulus–response or sensor-motoric level. At this level, both external and internal variations are considered as noise. Anything that is not controllable is considered as noise. However, when we go up to higher behavioral levels, such as rule-based and knowledge-based levels (Rasmussen, 1986), variations cannot be considered as noise any longer. Variations are integrating parts of the Human–Machine System. They are mostly caused by non-linearity. If automation at the skill-based level was correctly treated using automatic control theories and human engineering, automation at higher behavioral levels requires different approaches. Since we kept, often implicitly, the skill-based level approach for the automation of higher behavioral levels, it is not surprising that current highly automated systems are sometimes leading to unexpected events (see Fig. 3).

In fact, the term “automated” is no longer appropriate to denote such systems. It is more appropriate to talk about “software-intensive” complex systems.

A large amount of human factors studies has been carried out during the eighties and nineties, when automation drawbacks emerged, such as “ironies of automation” (Bainbridge, 1983), “clumsy automation” (Wiener, 1989), and “automation surprises” (Sarter, Woods, & Billings, 1997). When we look at these investigations and results today, we must recognize that technology maturity and maturity of practice were not considered, but are important factors that lead to a better definition of function allocation and finally automation (Boy, 2013). In particular, functions cannot be correctly allocated among humans and machines without a thorough identification of emerging cognitive functions, which lead to good design. Consequently, design is necessarily iterative, and supported by M&S with humans in the loop, especially prior to product delivery. Effectiveness of M&S means make HCD possible and efficient.

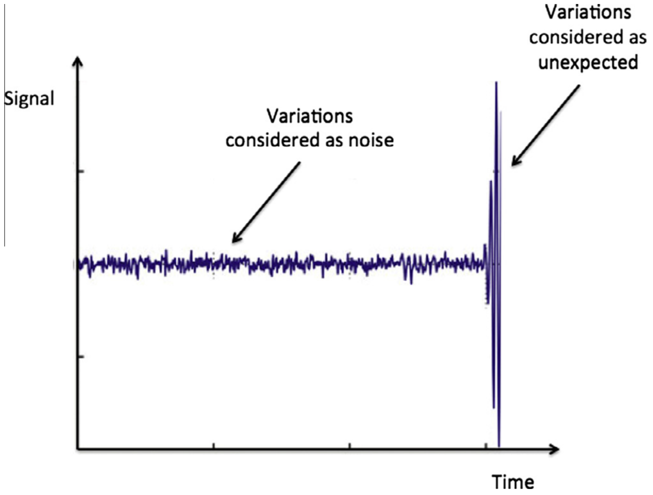

Fig. 3. Variations as noise and variations as unexpected.

2.3. Function allocation and emergent cognitive functions

For all these reasons, the cognitive function representation is very useful to support function allocation. Functions are not only allocated deliberately as what is being considered in engineering for a long time (Fitts, 1951). We cannot consider people as black boxes to be introduced in systems engineering diagrams. Even if we design a very simple system, people using it will necessarily introduce a tremendous amount of complexity. *Human-systems integration* (HSI) is a matter of discovering hidden properties and emerging behaviors coming from the interactions between people and technology, as well as among people themselves using this technology. We then need to identify the emergence of (cognitive) functions that arise from these interactions.

The only way to discover emerging cognitive functions at design time is to use modeling and simulation (M&S), and more specifically human-in-the-loop simulations (HITLS). This is why M&S, including HITLS, is a crucial discipline in HCD. M&S enables early usability and usefulness investigations and engineering. Bringing people to the Moon would not have been possible without M&S, even if it was very rudimentary at that time. M&S enables projection into possible futures and test them. That is very different from the classical predictive causal approaches based on past experience. Predictive approaches typically lead to short-term solutions,

Elapsed Time	Agents involved	Context	Triggering Preconditions	Goal/Role	Actions (dolist)	Abnormal Conditions (alerts)	Resources	Elicited Cognitive Functions

Fig. 2. Template of a CFA timeline analysis table showing a sample of the various entities being recorded, analyzed and elicited.

as approaches based on the assessment of possible futures enable the investigation of longer-term solutions. Both event-driven prediction and goal-driven projection are equally important.

3. What is human-centered design?

Human-centered design (HCD) of life-critical systems, as it is developed and taught at Florida Institute of Technology in the HCD Ph.D. program, can be defined by six major topics.

(1) HCD is strongly based on an expansion of cognitive engineering from single-agent models of cognition to multi-agent models of socio-cognitive interactions. For that matter, human and social sciences are needed to support understanding of HCD. Cognitive function analysis is at the center of this first topic.
(2) Life-critical systems (LCSs) include a large number of systems where people and machines interact with respect to three main principles, which are safety, efficiency and comfort. For that matter, comparison of various LCS domains is needed to better grasp these principles. LCS development requires creativity and design thinking.
(3) Contemporary human-centered design of life-critical systems (HCD-LCSs) cannot be dissociated from the evolution and constant development of advanced interaction media, e.g., the Internet and visualization techniques. For that matter, computer science and human–computer interaction are needed to guide design and development choices.
(4) LCSs are almost always complex in the sense of larger number of components, interaction among these components are highly non-linear. This is why mastering complexity analysis is so important. For that matter, complexity science, including chaos and catastrophe theories, is needed to better understand the nature of LCS complexity.
(5) Since technology usage induces the emergence of new organizations, organization design and management also need to be human-centered. For that matter, organization sciences and political science are also very useful.
(6) Finally, modeling and simulation (M&S) need to be extensively used to both foster creativity and rationalization of 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.

3.1. Inside-out engineering versus outside-in design

Most machines and systems have been developed *from inside out*, i.e., a kernel is developed, such as a car engine, and later on when the whole car is developed, we start to figure out how it will be driven. This was a necessary step because technology had to be created, most of the time from scratch, before being used. Only forty years ago, we were able to repair a car by ourselves when we knew about mechanics. Today, it is totally impossible because mechanics is not accessible any longer. We need to go to a specialized garage, and the technician will use a diagnostic system that will provide him/her with the pieces of equipment to be changed, and will give us a price for the overall repair in a few seconds. This means that we incrementally automated machines going from mechanical engineering to information technology (Fig. 4). This process is commonly called automation.

Fig. 4. Automation as the "old" shift from mechanical engineering to information technology: the inside-out approach.

Automation is usually thought as a layer on top of a mechanical system. We still design and develop cars from inside out, i.e., from mechanical parts to software components that enable using the car in safer, more efficient and comfortable ways. The user interface syndrome and necessity emerged over the years. Indeed, the user interface concept is only the result of this inside-out approach to engineering.

Increasingly-mastered knowledge of mechanical behaviors and processes and the evolution of *computer-aided design* (CAD) and *computer-aided manufacturing* (CAM) technology led to the concept of virtual prototypes (Coze et al., 2009). Virtual prototypes enable design teams to build models that are very realistic nowadays, and enable human-in-the-loop simulations. The Falcon 7X was entirely designed using this approach and "flown" before any hardware was physically developed. In addition, it is possible to virtually reverse-engineer (and further physically build) manufacturing buildings and tools to physically build the airplane. Everything comes from software and can be tested very early with appropriate people in the loop.

Today, we have more technology than we can afford to use. Design problems are very different. Technology parts are there, and the main issue is to integrate them. Integration is about creativity, continuous evaluation and validation. Design can be done *from outside in* (i.e., starting from user requirements and needs to technology integration, and eventually definition of new technology parts.) Human-centered design is at the heart of this outside-in approach.

The Google self-driving car is another example of the current revolution that brings us from information technology to graspable physical things. Google is a software company that developed a Web search engine. They also developed a fantastic way of providing maps and geographical directions to anyone on the planet Earth. They then mastered the environment for navigation purposes. It was tempting to design and develop robotic means that provide automated situation awareness and decision-making to a "tangible cognitive" car. The rest is history! They put wheels under the resulting computer! We already are in the field of tangible interactive objects. We now put hardware around software (Fig. 5). This new approach contrasts with the twentieth century approach that consisted in putting software into hardware.

3.2. Human-centered design is now possible!

Why is this at all possible and becomes real? Computer-based modeling and simulation methods and tools are incrementally developed and used. They are not new. CAD-CAM technology started to develop during the eighties to replace former drawing methods and physical tools (e.g., pencils, pens and paper). Today, they integrate dynamic simulations that enable the incorporation of potential users in the loop. This enables testing before anything is physically built. For example, professional pilots can fly an aircraft before it is built. Pilots fly a software tool, which is more realistic as we progress in time. This is a crucial paradigm shift. Instead of doing corrective ergonomics when a product is fully developed, we can now test realistic prototypes before anything is developed.

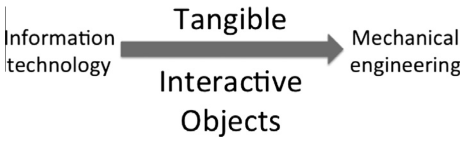

Fig. 5. Tangible interactive objects as the "new" shift from information technology to mechanical engineering: the outside-in approach.

We are shifting from technology-centered engineering to human-centered design. Of course, the aeronautical industry was already involved in HCD for a long time, practicing physical flight tests. This was very expensive. M&S enables much cheaper tests, and contribute to a better definition of fewer mandatory physical flight tests.

3D printing is certainly the latest technology breakthrough. Now, we can 3D print objects from CAD tools directly. These physical objects come from virtual prototypes and can be physically manipulated. We are definitely going from information technology to mechanical engineering. In addition, we are now able to combine these physical objects with software-based behaviors. For example, Wang et al. (2013) created a user-interactive electronic skin for instantaneous pressure visualization. Electronic skins can then be used to give some kinds of activities and behaviors to 3D printed objects. Consequently, these tangible interactive objects could be used to interact with other objects. The concept of tangible user interface was born in the field of human–computer interaction to denote the media that somebody uses to interact with digital information through the physical environment (Ishii, 2008). Horishi Iishi and his team at the MIT Media Lab developed what he called tangible bits (Ishii & Ullmer, 1997). Tangible bits are directly perceptible and easy-to-manipulate objects that give physical form to digital information. In this article, Iishi's initial concept is extended to large complex systems, and coined as "tangible interactive objects."

4. Combining human-centered design and systems engineering

4.1. Adapting the engineering V-model

Today, industry uses systems engineering as an interdisciplinary approach and means to enable the realization of successful systems (Haskins, 2011). The most popular engineering approach is the *V-model* (Pressman, 2009). The V-model is commonly used in industry to support life cycle of system production, going from design, development, manufacturing, certification and delivery. The V-model is grounded in software engineering. It has advantages such as its simplicity and usability.

The first leg of the V represents design and development. The second leg represents manufacturing, certification and delivery. Problem is that very little is done (money-wise) in the beginning (this explains the narrow first leg of the "red-color[1]" V in Fig. 6, which gets wider and wider on the second leg). Indeed, most efforts are concentrated in the second part of the V-model, with excessive amount of load close to the end to compensate "unexpected" events that often come from poor requirements defined in the beginning. Among other disadvantages of the V-model are its linear technology-centered nature that induces high rigidity and poor flexibility. It usually fails because users' needs and requirements are not well taken into account from the beginning of the process. Typically, technology-centered engineering (TCE) major efforts are brought during the second part of the V-model, and unfortunately towards the end to recover design flaws (Fig. 6).

It is not by asking people what they want that we define good high-level requirements for a product. It is by developing prototypes from the very beginning of the design process and involving potential users in the testing that we obtain good requirements. This is precisely what HCD can bring to systems engineering (Boy & Narkevicius, 2013).

Instead, HCD concentrates efforts during the first part of the V-model (the "blue-color" V in Fig. 6), and attempts to shape appropriate requirements that would lead to a successful product in the end. Product maturity is a matter of well-defined high-level requirements, continuous human-in-loop testing (by using modeling and simulation from the beginning of the design process), and refining both functional and structural architectures.

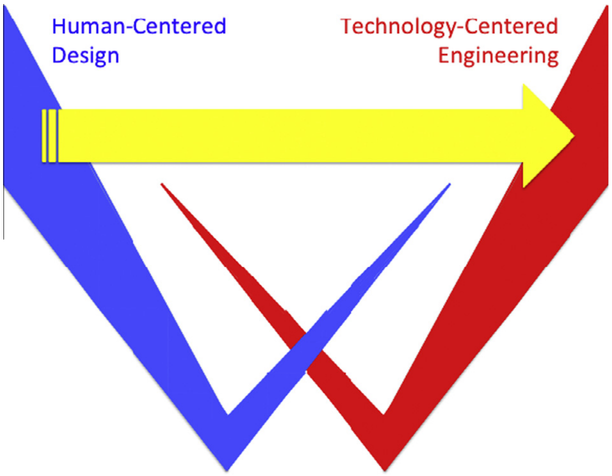

Fig. 6. Combining human-centered design and technology-centered engineering.

4.2. From the user interface paradigm to the TOP model

Up to now, Human factors and ergonomics (HFE) were taken into account at the end of the V cycle, i.e., when it was too late to modify the main concepts leading to the product. Therefore, the user interface and other artifacts such as user manuals, procedures and do-lists/checklists were developed to often compensate for design flaws, and adapt people to technology, as opposed to what HFE traditionally promotes. It is time to depart from this TCE approach to an HCD approach. TCE is based on the traditional positivist approach where we cut the world into pieces, engineer linear systems and reassemble them to make complicated machines. HCD is based on a phenomenological approach where we integrate models of future products from the beginning, test them in simulation and derive appropriate integrated designs that will be engineered later.

By introducing HCD from the very beginning of the design process, we break the traditional paradigm of the user interface, which is usually designed when the product is fully developed taking into account people using it.

In HCD, "systems" are commonly denoted as "agents", which can be people or software-based artifacts (i.e., tangible interactive objects as defined earlier in this paper). Agents are functionally defined as displaying cognitive functions, defined by their roles, their contexts of validity and supporting resources (Boy, 1998b). 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. There is a strong overlap between two approaches developed in isolation and that need to be associated, which are multi-agent systems (Salamon, 2011) and systems-of-systems (Clark, 2009). The evolution from automation to tangible interactive objects requires it.

Designing and developing such multi-agent systems requires an integrated model that involves Technology, Organization and People need to be considered in concert from the beginning (Fig. 7). This model is called the *TOP model* (Boy, 2013):

---

[1] For interpretation of color in Fig. 6, the reader is referred to the web version of this article.

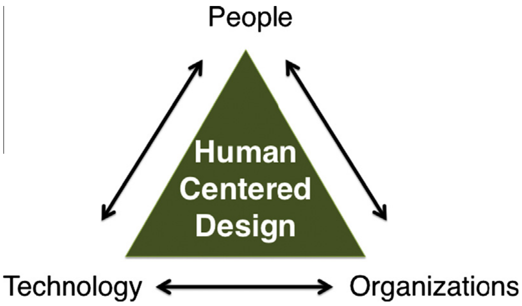

Fig. 7. The TOP model.

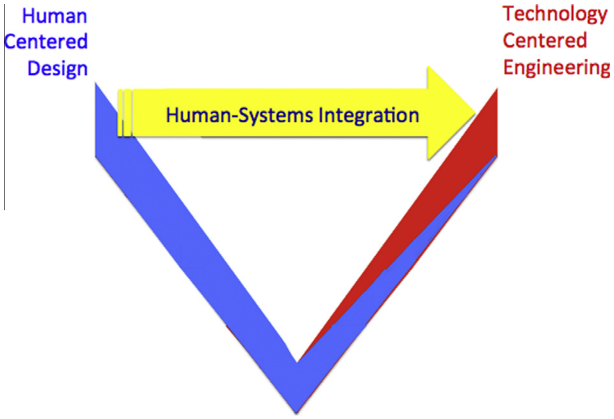

Fig. 8. Human-systems integration.

- Technology (i.e., the product being designed and developed).
- Organizations (i.e., the way the various actors work together using the product).
- People (i.e., the roles and jobs induced by the introduction and use of the product).

4.3. Defining human-systems integration

It is now clear that we need to merge human-centered design and systems engineering to achieve a better definition of *human-system integration* (HSI) (Fig. 8). Note that HCD should decrease the amount of load put on later Technology Centered Engineering (see Fig. 8 and compare to Fig. 6), as it is observed today (i.e., we need to pay the price to compensate poor requirements set up in the first place). In addition, experience and expertise will also decrease the load on the HCD part also.

Shifting from the traditional positivist approach of engineering (i.e., the materialistic builder approach) to a more phenomenological approach of design (i.e., the humanist architect approach), human-systems integration architects can no longer restrict their investigations using linear methods. They need to learn and use non-linear methods that come from complexity science, such as non-linear dynamical systems, attractors, bifurcations, fractals, catastrophes, and more (Mitchell, 2008). It is time to investigate variations as important information for human-systems integration, instead of considering them as noise and rejecting the parts of the Gaussian "bell" curve that we do not understand and do not fit our a priori assumptions. Complexity should be assessed through modeling and simulation with people in the loop. This is the only way to design for flexibility and accountability. Cognitive function analysis is a great method supporting such investigation effort, because it forces designers and engineers to formulate roles, contexts of use and resources required to accomplish prescribed tasks and realize desired activities (Boy, 1998b).

5. Looking for an organization model supporting HCD

Since HCD is now possible using information technology, it is crucial to better understand organizational issues, opportunities and constraints related to human-systems integration. Modeling and simulation, interconnectivity and advanced human–computer interaction already created the emergence of new practices and organizational setups. For example, we can now develop new aircraft systems on simulators that are very close to real-world aircraft cockpits interconnected to others, simulated air traffic control and real-time weather databases. Consequently, holistic approaches can be implemented and induce several paradigm changes such as shifting from single-agent to multi-agent approaches, developing cognitive function analyses and looking for different organizational working frameworks.

5.1. From single-agent to multi-agent approaches

For a long time, we focused on single-agent approaches to engineering (i.e., a human facing a machine). Human factors and ergonomics specialists modeled people, using control theory mathematical functions (quantitative analogs to skill-based behaviors, in Rasmussen's sense) and "if–then" production systems (symbolic representations of rule-based behavior) for example, as single entities connected to a machine that was modeled in the same way. Human–machine models were developed and some of them were very sophisticated, such as MESSAGE (Boy & Tessier, 1985) and MIDAS (Corker & Smith, 1993). These models were based on computer architecture analogs such as Newell and Simon's and Rasmussen's models (Newell & Simon, 1972; Rasmussen, 1983). Even if several agent models were assembled together, they were not numerous enough to qualify as multi-agent systems capable of generating some kinds of emerging behavior.

We needed to wait until the nineties to see a clear emergence of distributed cognition (Hutchins, 1995). Anthropologists and social scientists were needed to open this new kind of approach to human modeling. Distributed cognition enables the study of cognition at the individual level, but also at social and organizational levels. Cognition could not be considered as only internal to people, but also external and distributed among an organized set of agents. At the same time, the computer-supported cooperative work (CSCW) community started to develop (Grudin & Poltrock, 2012). This community contributed to develop different technologies that currently support collaborative activities. CSCW is deeply grounded in human–computer interaction (HCI) toward computer-mediated communication.

Other work efforts were made in the artificial intelligence community (Bradshaw, 1997; Ferber, 1999; Minsky, 1985). In this multi-agent systems (MAS) community, researchers develop computer models and simulations of agents interacting among each other. They can be reactive (event-driven) and/or intentional (goal-driven). Agents have roles. In MAS, agents are computer programs that enable autonomous execution of a task. They are usually able to perform some kind of reasoning. Russell and Norvig (1995) defined an agent as "anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors." The notion of autonomy of agents was introduced by Pattie Maes (1995).

CFA was developed under the influence of distributed cognition, CSCW and MAS, and is grounded on a multi-agent approach from the start. For example, an agent can be defined as a complex network of cognitive functions, which is itself a macro cognitive function. In addition, network of cognitive functions could be defined over a network of agents. This last property enables the representation of delegation, supervision, cooperation and coordination, and therefore is a good support for the identification of emerging cognitive functions. This is very interesting because it enables to model and simulate current highly automated Human–Machine Systems where management tasks have become predominant on top of progressively-vanishing control tasks. This is a matter of evolution.

Cognitive functions can be organized and form a high-level cognitive function. They can also be distributed among a group of agents (e.g., an agent may be leading a group of agents, delegating some tasks to others for example). CFA is extremely useful when we need to analyze a complex socio-technical environment where people and systems interact among each other. Designing TIOs, it is useful to define HSI complexity metrics for assessing properties of such environments (Boy & Grote, 2009). Properties could be resilience, stability, autonomy, authority and/or agent's accountability.

5.2. The Orchestra model

At the same time, our society has evolved. Our organizations are changing drastically. This is due to the influence of new technology on people and interaction among people. We cannot behave the same when we constantly use a smart phone compared to what we were doing before without it. In this specific example, smart phones contributed to replace goal-driven behavior by event-driven behavior (e.g., we used to have a list of items to buy at the grocery store, and we now use our smart phone to call home and check what is needed!). Our entire life is consequently organized differently. We then need to have better models to assess what we are doing in design and engineering if we really want to have a human-centered design approach. Up to now, our industrial companies and governmental agencies were organized with respect to a military model, typically top-down. Communication and information technology contributed to transversalize the various interactions among people in all types of organization. Consequently, there is an urgent need for orchestrating these new organizations. In addition, jobs and roles became more specialized (i.e., going from soldiers to musicians). The Orchestra model was developed to support this type of analysis and understand the resulting organizational evolution (Boy, 2013).

The Orchestra model supports CFA, providing a structure for the identification of both deliberative and emergent cognitive functions. This is very important to carry out effective HCD. For example, there may be disjunctions between the way organizations are still structured and the way people behave in them especially using new technology, which causes consistency and synchronization problems (e.g., the traditional military model imposes some practices that are incompatible with new practices already observed within the Orchestra model). We then need to find out the best co-adaptation of human and system cognitive functions iteratively. Again, modeling and simulation are excellent instruments for this kind of investigation and design.

6. Design, visualization and integration of new systems

In nuclear and aerospace industries for example, most control rooms have been designed a few decades ago, and are still in service. They include instruments and controls that can be quite old. In addition, their design corresponds to the philosophy of the time they were built. Today, we have new kinds of technology that is almost entirely digital. Furthermore, most of us use this technology in our everyday life (e.g., tablets, GPS and highly connected personal computers, sometimes hidden at the heart of appliances that we use, such as smart phones). It is then expected that new control rooms will be equipped with at least equivalent technology. Old control rooms involve many people. Automation contributed to decrease the number of human operators and changed the nature of their work. The development of tangible interactive objects will continue to change the nature of interactions among people and systems as well as among people themselves interacting through systems. Unlike the old approach (20th century technology-centered engineering, incorporating information technology into mechanical things), the new one (21st century human-centered design, making tangible interactive objects from virtual prototypes) enables to take into account and better understand people's roles from the early stages of design and development. Now, CFA can typically support function allocation toward effective human-system integration by enabling both rationalization and evaluation of progressive mockups and prototypes until a satisfactory result is found.

Today, we have large multi-touch interactive surfaces that can equip walls and tables (Kubicki, Lepreux, & Kolski, 2012; Müller-Tomfelde, 2010). It is then convenient to think about using this kind of technology to equip control rooms, where people will interact not only with technical parameters of the processes being controlled, but also with other personnel who are not necessarily co-located. Therefore, it is crucial to better understand how to integrate new interactive technology to enable personnel to interact with both humans and systems. Such integration will provide a tremendous power. The problem is eliciting and orchestrating the various cognitive functions involved in the control of processes involved. This is a matter of organization design and management. At this point, we can see that human-centered designing the control room brings to the front new requirements for the way processes will be managed in the future. This is designing from outside in, i.e., from the purposes and operations to means and engineering.

In addition, there are processes that were handled mechanically in the past and therefore required humans to manipulate physical machines. Today, robotics and visualization software enable us to process data and information instead. We are progressively moving from control of machines to management of systems. This shift requires understanding of how personnel jobs are affected and how they will potentially change (i.e., looking for emerging cognitive functions). This is why CFA is required. Safety–critical systems industries, such as the nuclear industry, extensively developed experience feedback to better understand, fix system failures and to address human errors. In some cases, it worked so well that they have compiled so many regulations, leading to a huge amount of procedures, that it is now difficult for human operators to handle procedures correctly because of their constantly increasing number (Boy & Schmitt, 2012). Consequently, it is now urgent to redesign not only the control room, but also the control and management system as a whole. Experience feedback is part of the human-centered design process, but it should be done much more in advance during the design process using modeling and simulation. The more it is done late in the design process, the more it will involve extra costs to integrate. When it is only done during operations, it typically leads to unexpected modifications, large extra labor costs, and significant impacts to schedule.

Today, we are able to visualize large amount of data. The visualization field of research is developing fast and should be giving great results in the near future (Bederson & Shneiderman, 2003; Tufte, 2001). We are already able to visualize complex data that provide appropriate insights and explicit representations of crucial phenomena. We are able to visualize natural scenes (e.g., Google Earth scene of a catastrophic accident or event, such as an earthquake) and superimpose meaningful artifacts on it (e.g., recovery procedures, reports from emergency teams, evolution diagrams and calculated trends, and various kinds of parameters integrated into physical schematic representations). Such integrated types of visualization provide meaning to human operators.

Designing a control room offers the opportunity to integrate new technological pieces. This integration involves synthetic minds and creativity. Current technology enables us to incrementally integrate and go from prototypes to real systems in a straightforward manner. We need to think in terms of individual management and control, but also in terms of communication, cooperation and coordination. Interactive technology provides this kind of support to both individual and collective interaction. It is important to remember that we are always designing with our past experience, with tools of today for people who will operate in the future. A complex plant is not designed to be disposable in a few months; it is developed for a few decades. Who will be the human operators twenty years from now? Nobody can answer this question. We just can guess and try. We need to better understand the trend of the possible evolution of the TOP model (technology, organization and people). This is why current work on the evolution from the military model to the Orchestra model is important. Other alternative models are also welcome.

7. Discussion

The twentieth century saw the evolution from hardware to software, from complicated machines to complex systems. Twenty-first century complexity comes from the interconnections among systems and people (e.g., transportation systems and Internet). Related contemporary research topics are focused on emergent properties of systems and multi-agent interactions. This is why complexity science has to be further developed in order to understand this evolution.

As an example, aviation has evolved since Clement Ader's flying machine, *Eole* (Boy, 2013, page 90). In the beginning, airplanes were only mechanical objects, which were based on two main attributes: thrust and lift. Engines take care of the thrust, and horizontal surfaces (e.g., the wings) take care of the lift. Over the years, mechanical engineering developed better solutions that improved operationalization of these two attributes. However, there were three main issues that governed the evolution of commercial aviation during the twentieth century: safety, efficiency and comfort. The number of instruments in cockpits increased to about 600 for Concorde (for three crewmen cockpits). Until then they were electro-mechanical instruments. Cathode ray tubes and higher-level automation contributed to decrease the number of instruments in the cockpit of commercial airplanes, but not the number of parameters and functions. Aircraft avionics became more electronic and then more software intensive leading to the concept of an "interactive cockpit" (i.e., the term "interactive" refers to human–computer interaction and more specifically the use of a pointing device). Pilots now interact with computers and not with mechanical structures directly. How did we come to this point?

For a long time commercial aircraft pilots controlled the attitude (pitch and roll) of the aircraft using a yoke, also called a control column, which was directly mechanically connected to the ailerons and roll axis (for rotation) and the elevators and pitch axis (for fore and aft movement). When we introduced fly-by-wire control systems, the yoke was replaced by a side-stick on some aircraft during the eighties. Autopilots were introduced onboard aircraft a long time ago (around the 1930s). The autopilot was designed as a single agent system automating the control of the trajectory around the center of gravity (Cg) of the aircraft, one parameter at a time, with a time constant of around 0.5 s (Fig. 9).

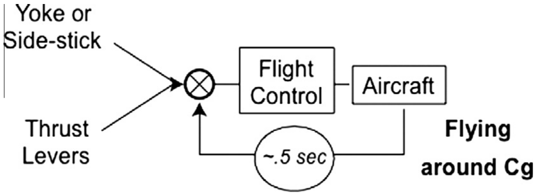

Fig. 9. Trajectory control automation (Tarnowski, 2006).

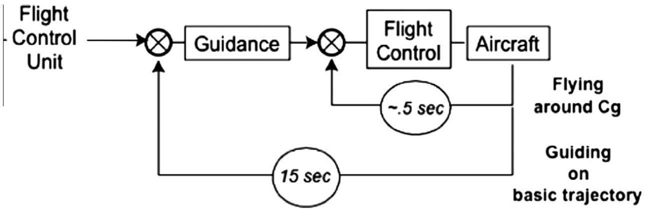

Fig. 10. Guidance automation (Tarnowski, 2006).

Then, guidance was automated, integrating digital autopilots and auto-throttle (Fig. 10). High-level modes of automation were introduced with a time constant of around 15 s. This was a major evolution.

The next step was the automation of navigation by introducing the flight management systems (FMS) with a time constant of around 1 min. This was a radical change that transformed the job of pilots from control to management. They had to program the flight plan using a new cockpit device, the control and display unit (CDU). Computers were onboard to stay; glass cockpits were definitely born (Fig. 11).

The latest loop, which is difficult to describe using a classical control theory diagram, is the "automation" of the air traffic management (ATM). The datalink (i.e., digital communication instead of very high frequency technology) was introduced between the flight deck and the air traffic control. We moved into a new era of automated data and information management, where the time constant is around 10 min (Fig. 12). In addition to the radio management panel (RMP), pilots are now using a datalink control and display unit (DCDU).

Of course, this evolution continues, and we are now working on new systems that will display the whole traffic around the aircraft (similar to radar screens of air traffic controllers), and so on. This evolution is not only due to the production of new systems because technology is available, but it is also mandatory to handle the evolution of the airspace. Indeed during the last three decades, yearly average air traffic increase over western countries is 4.5%. Some airports are saturated over 100%. Air traffic controllers may have to control more than 100 aircraft per hour. Air traffic density does not stop to increase. This leads to a situation that is difficult, and soon impossible, to manage with current methods and organizations. Using the TOP model, solutions are threefold: (1) automation of the sky; (2) re-organization of the sky; and (3) definition of new jobs. In fact, "automation of the sky" is now an old paradigm that should be replaced by clearer definitions of airspace TIOs. Managing the density of aircraft in the sky is a matter of complexity science, where inter-connectivity among aircraft should be made explicit instead of implicit like it is the case today. Modeling and simulation is crucial to find out best solutions, because these solutions are very likely to progressively emerge from M&S, as attractors. Once these solutions will be found, they will need to be implemented into physical systems, which will become airspace TIOs.

Re-organization of the sky is a matter of modeling possible futures, which need to further studied. The Orchestra model was consolidated during the PAUSA project, which investigated

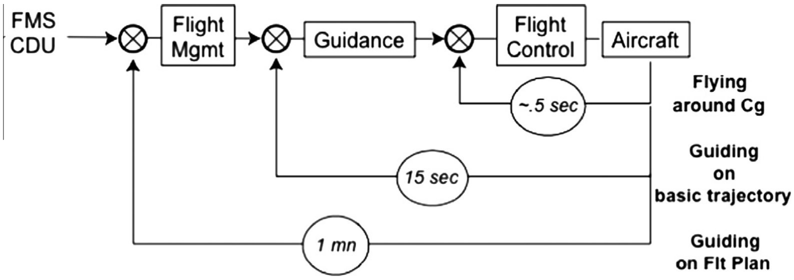

Fig. 11. Navigation automation (Tarnowski, 2006).

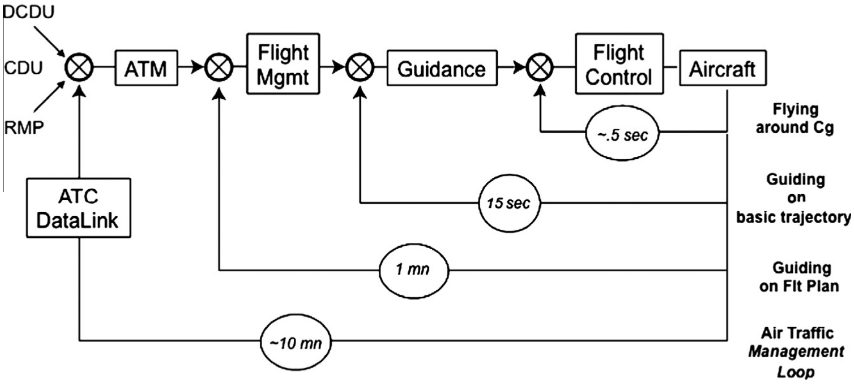

Fig. 12. "Automation" of the air traffic management (Tarnowski, 2006).

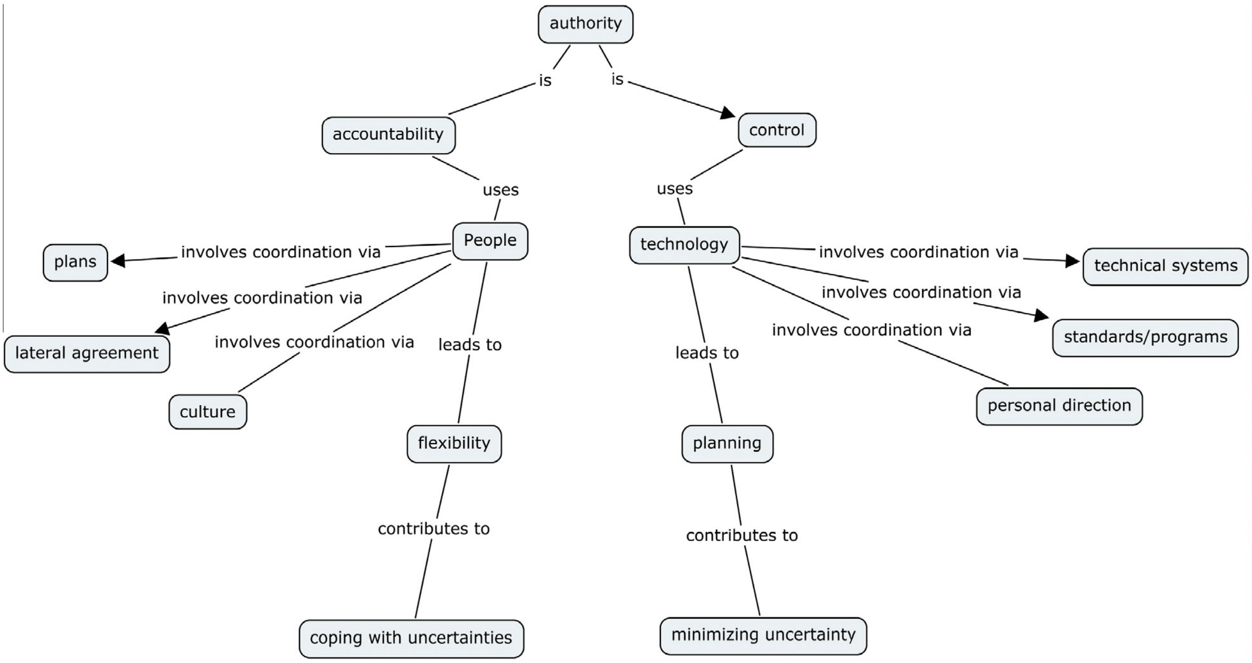

Fig. 13. Authority concept map.

authority sharing in the airspace (Boy, 2009; Boy & Grote, 2009). Authority is both a matter of control and accountability (Fig. 13). When we only take the control side of authority, we deal with technology that leads to planning, itself contributing to minimize uncertainty. However, planning increases rigidity (i.e., the more we automate the more we rigidify). The accountability side of authority deals with people who provide flexibility and cope with uncertainty. HCD takes into account both sides. The former involves rigid coupling; the later involves loose coupling. Of course, uncertainty is minimized using technology when it is mature and well mastered in its context of use (i.e., corresponding technology-related cognitive functions are well identified, understood and experienced). Similarly, people cope well with uncertainty when they have enough knowledge and skills to handle socio-technical situations (i.e., corresponding cognitive functions are properly learned and operationally mastered).

Definition of new jobs is inevitable. As in many industrial sectors, the progressive increase of the number of actors and software-intensive systems induces a necessary shift from control to management. However, this can be done when both technology maturity and maturity of practice are both acceptable. As already commented above, this is a matter of identification and control of emerging cognitive functions. Aeronautics follows this path by moving from air traffic control (ATC) to air traffic management (ATM). It was the same during the eighties when we developed the glass cockpits and the fly-by-wire on commercial aircraft; pilots moved from controlling handling qualities to managing systems (i.e., from flying to managing the flight). This was because we developed automated systems that are now taking care of low-level control tasks; pilots have now higher-level flight management tasks. Being an airline pilot today requires systems management knowledge and skills, as well as handling quality skills when systems go wrong. The major difference with the past is the fact that safety tremendously increased. Back to the ATM problem, we now need to better understand what will become the new jobs in the airspace. There will be shifts of authority among humans and systems, as well as between flight decks and ground stations.

This discussion was deliberately carried out on a concrete example to show the importance of the concurrent evolution of technology, organizations and people (the TOP model). Complexity analysis requires new methods and tools that can be provided by modeling and simulation with humans in the loop. We also need to better understand the evolution in terms of life-critical systems where software is now integrated in almost every bit of our lives. In addition, instead of pilling up layers of automation incrementally satisfying short-term needs, information technology and M&S enable top-down HCD leading to appropriate TIOs satisfying long-term requirements.

8. Conclusion and perspectives

Our society has gone very far in the production of technology and will continue to do so. However, it is time to realize what we are doing to our planet and our societies. This epistemological essay on automation was started to better understand the nature of the evolution toward the integration of software in our lives. It is far from being completed, but others can take the torch and further develop this preliminary work. Since we have chosen to develop more technology, it is important to do it well, i.e., with people at the center.

Software integration into our lives causes incremental dematerialization. Dematerialization has become a common ground for most of us on this planet (Diamandis & Kotler, 2012; Kurzweil, 2005). We can bank, book a flight and get a boarding pass using our cell phone. No paper is needed any longer in a large number of operations. However, we are still human beings with physical needs. We need to grasp things that make physical sense. Living in the artificial and virtual all the time is not enough for most of us. This is why we need to re-materialize our lives by thinking in term of tangible interactive objects. Software is good for improving safety, efficiency and comfort. However, we need to incrementally redefine what we mean by safety, efficiency and comfort, because any time we developed technology that effectively improved our lives, we looked for more to do using this technology. This is catch 22[2]!

Will we fly commercial airplanes without pilots in the future? The answer is very probably yes! We are able to build drones that fly autonomously. The question is: will people accept to be passengers in such drones? The answer is probably no in the short term. However, it is quite possible to have such transportation means in a longer term; acceptability is a matter of reliability, maturity and responsibility. Today, pilots have the responsibility of flight safety, together with air traffic controllers and aeronautical engineers. With drones, there will be new jobs to be created, such as monitoring operators and real-time flight planners. The authority issue has to be investigated and redefined among the actors. Today, we are incrementally pushing automation toward an asymptote (that is full automation), which cannot be reached with our current ways of thinking. We need to drastically move toward full multi-agent modeling and simulation that would provide the requirements for such an automation of the sky, as already explained.

Human-centered design of tangible interactive objects is certainly a good way to go. Consequently, we need to better focus on the relationships and evolutions of the three majors entities that are Technology, Organizations and People. The Orchestra metaphor was proposed to support the TOP model. We need to look for creativity, as synthesis and integration, and maturity, as stabilization of technology use in our lives. We need to study the increasing complexity of our societies, combining standardization (linearization) and singularities (non-linear bifurcations and catastrophes in Thom's sense). This is the price to pay to better understand and act correctly in the design of life-critical systems and the kind of education that goes with it.

Acknowledgements

This article is based on discussions and work with some of my former colleagues such as Gudela Grote, Etienne Tarnowski, and Jean Pinet. Thanks all. Many thanks also to Jarrett Clark, Ondrej Doule, Nicholas Kasdaglis, Rhonda Lyons, Jean-Paul Pinelli, Lucas Stephane and Tan Wei for reviewing this article and providing me with astute comments. Gratitude also goes to anonymous IFAC reviewers of this article, and Janos Gertler, the editor of this ARC journal issue.

References

Bainbridge, L. (1983). Ironies of automation. *Automatica, International Federation of Automatic Control, 19*(6), 775–779. Pergamon Press.

Bederson, B. B, & Shneiderman, B. (2003). *The craft of information visualization: Readings and reflections*. Morgan Kaufmann. ISBN: 1-55860-915-6.

Billings, C. E. (1991). *Human-centered aircraft automation: A concept and guidelines*. (NASA technical memorandum 103885). Moffett Field, CA: NASA-Ames Research Center.

Boy, G. A. (1987). Operator assistant systems. *International Journal of Man–Machine Studies, 27*, 541–554. and Cognitive engineering in dynamic worlds, G. Mancini, D.D. Woods, & E. Hollnagel, Ed., Academic Press, London.

Boy, G. A. (1991). Indexing hypertext documents in context. In *Proceedings of the ACM Hypertext'91 conference, San Antonio*, Texas, December, also in the ACM Digital Library.

Boy, G. A. (1998a). Cognitive function analysis for human-centered automation of safety-critical systems. In *Proceedings of CHI'98, the ACM conference on human factors in computing systems, Los Angeles, USA* (pp. 265–272). New York: ACM Press. ISBN:0-201-30987-4.

Boy, G. A. (1998b). *Cognitive function analysis*. CT, USA: Greenwood/Ablex. ISBN:9781567503777.

Boy, G. A. (2009). The Orchestra: A conceptual model for function allocation and scenario-based engineering in multi-agent safety-critical systems. In *Proceedings of the European conference on cognitive ergonomics*, 30 September–2 October, Otaniemi, Helsinki area, Finland.

Boy, G. A. (Ed.). (2011). *Handbook of human–machine interaction: A human-centered design approach*. UK: Ashgate.

Boy, G. A. (2013). *Orchestrating human-centered design*. 978-1-4471-4338-3. UK: Springer.

Boy, G. A., & Grote, G. (2009). Authority in increasingly complex human and machine collaborative systems: Application to the future air traffic management construction. In *The proceedings of the 2009 international ergonomics association world congress*, Beijing, China.

---

[2] The term "catch-22" denotes a paradoxical situation from which an individual cannot escape because of contradictory rules (Random House Dictionary, 2012). It expresses the absence of control.

Boy, G. A., & Platt, D. (2013). A situation awareness assistant for human deep space exploration. In *HCI international conference*. Las Vegas, Nevada, USA, July. Also on the ACM Digital Library (http://dl.acm.org). Also in Kurosu, M. (Ed.), *Human–computer interaction, Part IV, HCII 2013, LNCS 8007* (pp. 629–636). Heidelberg: Springer.

Boy, G. A., & Tessier, C. (1985). Cockpit analysis and assessment by the MESSAGE methodology. In *Proceedings of the 2nd IFAC/IFIP/IFORS/IEA conf. on analysis, design and evaluation of man–machine systems*, September 10–12, Villa-Ponti, Italy (pp. 73–79). Oxford: Pergamon Press.

Boy, G. A (2002). Theories of human cognition: To better understand the co-adaptation of people and technology, in knowledge management, organizational intelligence and learning, and complexity. In L. Douglas Kiel (Ed.), *Encyclopedia of Life Support Systems (EOLSS), Developed under the Auspices of the UNESCO*. Oxford, UK: Eolss Publishers. <http://www.eolss.net>.

Boy, G. A, & Ferro, D. (2003). Using cognitive function analysis to prevent controlled flight into terrain. In Don Harris (Ed.), *Chapter of the human factors and flight deck design book*. UK: Ashgate.

Boy, G. A., & Narkevicius, J. (2013). Unifying human centered design and systems engineering for human systems integration. In M. Aiguier, F. Boulanger, D. Krob, & C. Marchal (Eds.), *Complex systems design and management*. UK: Springer. ISBN-13:978-3-319-02811-8.

Boy, G. A., & Schmitt, K. A. (2012). Design for safety: A cognitive engineering approach to the control and management of nuclear power plants. *Annals of Nuclear Energy*. Elsevier. Available online.

Bradshaw, J. (Ed.). (1997). *Software agents*. Cambridge, MA: MIT Press/AAAI Press.

Clark, J. O. (2009). System of systems engineering and family of systems engineering from a standards perspective. In *3rd Annual IEEE international conference on systems engineering*. Vancouver, BC, Canada. ISBN:978-1-4244-3462-6.

Corker, K. M., & Smith, B. R. (1993). An architecture and model for cognitive engineering simulation analysis: Application to advanced aviation automation. In *The proceedings of the AIAA computing in aerospace conference*, October, San Diego, CA.

Coze, Y., Kawski, N., Kulka, T., Sire, P., Sottocasa, P., & Bloem, J. (2009). *Virtual concept: Real profit with digital manufacturing and simulation*. The Netherlands: Dassault Systèmes and Sogeti. LINE UP Book & Media. ISBN:978 90 75414 25 7.

Diamandis, P., & Kotler, S. (2012). *Abundance: The future is better than you think*. Simon and Schuster Digital Sales Inc.

Ferber, J. (1999). *Multi-agent system: An introduction to distributed artificial intelligence*. London, UK: Addison-Wesley.

Fitts, P. M. (1951). *Human engineering for an effective air navigation and traffic control system*. Columbus, OH: Ohio State University Foundation Report.

Grudin, J., & Poltrock, S. (2012). Taxonomy and theory in computer-supported cooperative work. In Steve W. J. Kozlowski (Ed.), *The Oxford handbook of organizational psychology, two-volume set (Oxford library of psychology)*. USA: Oxford University 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.

Hutchins, E. (1995). How a cockpit remembers its speed. *Cognitive Science, 19*, 265–288.

Ishii, H. (2008). The tangible user interface and its evolution. *Communications of the ACM, 51*(6).

Ishii, H., & Ullmer, B. (1997). Tangible bits: Towards seamless interfaces between people, bits and atoms. In *The proceedings of human factors in computing systems: CHI 97*. Denver, CO: ACM Digital Library.

Kaptelinin, V. (1995). Designing learning activity: A cultural-historical perspective in CSCL. In *Proceedings of the computer supported cooperative learning (CSCL'95)*. Bloomington, IN: Indiana University.

Kubicki, S., Lepreux, S., & Kolski, C. (2012). RFID-driven situation awareness on TangiSense, a table interacting with tangible objects. *Personal and Ubiquitous Computing, 16*(8), 1079–1094.

Kurzweil, R. (2005). *The singularity is near*. Penguin Group.

Leont'ev (1981). *Problems of the development of the mind*. Moscow: Progress.

Maes, P. (1995). Artificial life meets entertainment: Life like autonomous agents. *Communications of the ACM, 38*(11), 108–114.

Mark, W. (1999). Turning pervasive computing into mediated spaces. *IBM Systems Journal, 38*, 677–692.

Minsky, M. (1985). *The society of mind*. New York: Simon and Schuster.

Mitchell, M. (2008). *Complexity: A guided tour*. New York: Oxford University Press.

Müller-Tomfelde, C. (Ed.). (2010). *Tabletops – Horizontal interactive displays. Human–computer interaction series*. Springer. ISBN:978-1-84996-113-4.

Newell, A., & Simon, H. (1972). *Human problem solving*. Englewood Cliffs, NJ: Prentice Hall.

Pressman, R. (2009). *Software engineering: A practitioner's approach* (7th ed.). McGraw-Hill. ISBN-13:978-0073375977.

Rasmussen, J. (1983). Skills, rules, and knowledge; signals, signs, and symbols, and other distinctions in human-performance models. *IEEE Transactions Systems, Man, and Cybernetics, SMC 13*(3), 257–266.

Rasmussen, J. (1986). *Information processing and human–machine interaction: An approach to cognitive engineering*. 0444009876. New York, NY, USA: Elsevier Science Inc.

Russell, S. J., & Norvig, P. (1995). *Artificial intelligence: A modern approach*. Englewood Cliffs, NJ: Prentice Hall.

Salamon, T. (2011). *Design of agent-based models*. Czech Republic: Tomas Bruckner, Repin-Zivonin.

Sarter, N. B, Woods, D. D., & Billings, C. E. (1997). Automation surprises. In G. Salvendy (Ed.), *Handbook of human factors & ergonomics* (2nd ed. Wiley.

Sheridan, T. B. (1992). *Telerobotics, automation, and human supervisory control*. 0 262 19316 7. Cambridge, MA: MIT Press.

Sheridan, T. B. (Ed.). (1997). *Perspectives on the human controller*. Mahwah, NJ: Erlbaum. ISBN: 0 8058 2190 2.

Sheridan, T. B. (2002). *Humans and automation: System design and research issues*. 978-0-471-23428-9. NY: John Wiley and Sons.

Tarnowski, E. (2006). *The 4-Loop approach to ATM. Keynote speech at the international conference on human–computer interaction in aeronautics (HCI-Aero'06)*, Seattle, USA.

Thom, R. (1983). *Paraboles et catastrophes*. Éd. Champs Flammarion, no 186.

Tufte, E. R. (2001, First ed. 1983). The visual display of quantitative information (2nd ed.). Cheshire, CT: Graphics Press. ISBN: 0-9613921-4-2.

Wang, C., Hwang, D., Yu, Z., Takei, K., Park, J., Chen, T., et al. (2013). User-interactive electronic skin for instantaneous pressure visualization. *Nature Materials, 12*, 899–904. http://dx.doi.org/10.1038/nmat3711.

Weiser, M. (1991). The computer for the twenty-first century. *Scientific American, 265*(September), 94–104.

Wiener, E. L. (1989). *Human factors of advanced technology ("glass cockpit") transport aircraft* (NASA contractor report no. 177528). Moffett Field, CA: NASA Ames Research Center.

Guy A. Boy, Ph.D., is University Professor and Director of the Human-Centered Design Institute and Ph.D. & Master's Programs at the Florida Institute of Technology, IPA Chief Scientist for Human-Centered Design at NASA Kennedy Space Center and a Senior Research Scientist at the Florida Institute for Human and Machine Cognition (IHMC). He is member of the Scientific Committee of the SESAR program (Single European Sky for Air Traffic Management Research). He was the Chair of the 2012 ISU (International Space University) SSP (Space Studies Program) FIT/NASA-KSC local organizing committee. He is Adjunct Professor at the École Polytechnique in Paris (Comasic Master). He was the President and Chief Scientist of the European Institute of Cognitive Sciences and Engineering (EURISCO). He co-founded EURISCO in 1992, and managed it since its creation to its closing in 2008. Engineer and cognitive scientist, he received his Doctorate in 1980 from the Ecole Nationale Supérieure de l'Aéronautique et de l'Espace (ISAE-SUPAERO: The French Aerospace Institute of Technology), his Professorship Habilitation (HDR) from Pierre and Marie Curie's University (Paris VI), and his Full Professorship Qualifications in Computer Science and Psychology. Boy actively participated to the introduction of cognitive engineering in France and its development worldwide. He was the co-founder in 2004 of the Ecole Nationale Supérieure de Cognitique (ENSC), a cognitive engineering program at the University of Bordeaux. He is the editor the French handbook of cognitive engineering, and the Handbook of Human–Machine Interaction (Ashgate, UK) in 2011. His most recent book is Orchestrating Human-Centered Design (Springer, UK, 2013). Elected Fellow of the Air and Space Academy in 2006, he is a senior member of the ACM (Executive Vice-Chair of ACM-SIGCHI from 1995 to 1999) and INCOSE.