ADAPTIVE TECHNOLOGIES: USING A BIOLOGICAL EVOLUTION FRAMEWORK
Position Paper presented at IFIP WG 8.6 Working Conference
Guimaraes, Portugal
June 5, 2017
Eleanor Wynn
Ronin Institute
Portland OR

ABSTRACT

The working group is facing a transition. Its original mission of diffusion of information technology has been absorbed by historical developments in computer systems. But the adaptation portion can be addressed by shifting focus from the push of diffusion to the pull of emergence (Hagel et al., 2012) in a complex system. No longer compartmentalized artifacts, computer systems are sufficiently evolved and integrated with social systems that innovation, robustness and adaptation can be modeled on evolution in biological systems using complexity science constructs. AIs including Deep Learning and Model-Free Methods (Anderson) already possess the capabilities of self-evolution. The dimensions of intention and meaning largely absent from biological systems can be provided by dynamic social theories like actor networks (Latour) and assemblage theory (DeLanda). The position paper will list biological constructs useful for understanding adaptiveness. This allows us to view users, developers, and computer systems themselves as agents interacting within a technological environment to produce emergent outcomes.

INTRODUCTION

As we contemplate a new direction for IFIP WG 8.6, it is fitting to consider its original title and purpose as an historical artifact. When the working group was formed, there were pressing issues of adoption of new technologies. The change from relatively stable but slower practice-based physical or paper systems to electronic ones required a major shift in skills and perspective, a shift that had cascading effects as ever-new scopes for technology rapidly succeeded early developments. Issues of the day were “resistance to change”, and questions about whether managers would ever use keyboards. Indeed, in the early 80s, the term “keyboarding” was deliberately used to replace the term “typing”, to soften the blow to status and gender pride that typing might mean for managers, at the time mostly men. Things are different now. Persuading people to use computing tools is not an issue. The early electronic systems changed slowly relative to today’s rate; applications, tasks, and networks were highly compartmentalized. Programs didn’t interact across an enterprise; networks were in-house only, and many were proprietary, supporting a limited number of vendor terminals or PCs. In Labyrinths of Information, Ciborra (2002) describes the bricolage of legacy systems built up as capabilities were added one at a time, while core systems remained stable. Biskup and Kautz (1992) pointed out contradictions in stage models of technology evolution, positing a continuous process that can be realized today.

Systems that were a set of incompatible fiefdoms run by a small set of programmers who alone knew how to make them work, patch them or adapt them were subject to the fragility of gridlock, lack of fit, and duplicate effort. Users reflected these incompatibilities in their preferences. An inventory of tools corporate employees used in 2005 to share information (Chudoba et al.) found an archaeology of applications that reflected either a generation or the discipline of the employee: e.g. Excel for accounting and planning people, PowerPoint and Word for sales, regardless of content. Notes were taken in any one of the applications (Jensen et al.). Even in contemporary desktop applications, legacy concepts remained as fragmented application domains with an intended use based on past practice.
Many intermediate game changers paved the way to the more open and fluid systems in progress today: open protocol networks, world wide web access, html, mobile platforms, and graphics among others. The commodification of internet shopping brought massive numbers of users online, facilitated by sophisticated graphics, efficient shopping cart and payment experience, and price competition. Another watershed was the smart phone, with its portability, integration of telephone, camera, email, and proliferation of tailored apps. In short, based on these integrative capabilities, usage spread virally, and demand pull at some point outpaced diffusion push. Now the dilemma is how to put the brakes on runaway concepts far beyond our conventional understanding, applications of computing that rival the most imaginative science fiction and push the boundaries of morality and reality itself.

Core capabilities now include Deep Learning, virtual reality, electronic implants and total connectivity in IoT. Futurists and transhumanists (Kurzweil) entertain the idea of uploading brain contents to continue acting in the mode of Daemon (Suarez), where a deceased computer scientist continues acting long after his death in a Matrix-like network. Not all these inventions will take hold, but those that do take hold are taken up virally and with little help from their creators beyond the genius of the versatile platforms they are built on. Diffusion is accomplished by generativity of a platform, plus access and modularity.

**TIME FOR A NEW MODEL**

So, what is a grand, generative, extensible, and modular theme the Working Group can foster with its new focus? The risks of systems have shifted from internal incompatibilities to external permeability and misuse. The legacy banking systems that Ciborra et al. analyzed had a layered or laterally aggregated archaeology of capabilities within an organization; now we have diversity and inconsistency among functionalities, and potential catastrophic failures as systems become more conjoined without being designed as adaptive. The problems of pasted-up systems have been replaced by the problems of very large scale hackability and technological overreach. Inconsistency of security methods, naivete of users, proliferation of malware types, and malicious intent compound the problem. At the same time, the capabilities of AI, via model-free methods (Anderson) are awe-inspiring, frightening, and limiting all at once. A system that can beat a Go master in a match, with its own non-human intuition, is a wonder. Another system that collects and stores all communications and then looks for anomalies or intelligence on every citizen, is a potential horror, in its motivation, its analysis methods, its potential uses.

Another key distinction from the past is the eroding boundary between technical and human systems. Social networking technologies raise questions about whether computer-mediated activity is “real life” (Jurgenson 2011). There are fervent advocates for implants and human body augmentation, life extension, the singularity, and superintelligence that replaces human intelligence, not only in faster computing capacity, but in alternate approaches to problem-solving and intuition. Often these programs are invented by people with a techno-utopian focus oblivious to ethics, goals, or the nature of human intelligence and consciousness, even as they take on the task of transforming our techno-social world. The problem of robots replacing the workforce is simple by comparison. Implied definitions of consciousness and of reality accompany the whole endeavor. How do humans adapt or confront this?

These developments require strong multidisciplinarity, good philosophical underpinnings, and a motivation towards the benefit of society. In fact, they require a working group. There must be a co-evolution of human insight and technological capability. There are competing niches, contradictory trends and capabilities, different markets, audiences, and policy structures. We need the research to advise those policy structures and the practices, standards, and insights required to maintain non-totalitarian, human
BIOLOGICAL SYSTEM PROPERTIES

What are the properties of a system that resembles a biological evolutionary system? A complex system is self-organizing. It isn’t planned, and its outcomes are emergent. Complex systems arise from the interaction of agents following small localized rule sets. There is no top-down design. Diversity of agents is a key property of a complex system, as diversity contributes to the fit with the environment. As environmental conditions change, a self-organizing system adapts and evolves by means of mutation or invention in agents, by network and niche structure, and by the interactions that take place in response.

Krakauer, in talks at the Santa Fe Institute, has differentiated invention from innovation in this context. In nature, invention happens constantly, and most of nature’s inventions (mutations) fail. Innovation occurs when an invention creates a new ecosystem niche, or displaces an old one. Achieving this status doesn’t mean the innovation is necessarily the best standalone solution. Niches may be occupied—invention too late; or the invention can come too soon for there to be coevolutionary factors. Krakauer’s research agenda suggests further constructs for evolution and adaptation of information systems:

- molecular logic of signaling pathways, the evolution of genome organization (redundancy, multiple encoding, quantization and compression), robust communication over networks, the evolution of distributed forms of biological information processing, dynamical memory systems, the logic of transmissible regulatory networks (such as virus life cycles) and the many ways in which organisms construct their environments (niche construction)...

Many of these areas are characterized by the need to encode heritable information (genetic, epigenetic, autocatalytic or linguistic) at distinct levels of biological organization, where selection pressures are often independent or in conflict. Furthermore, components are noisy and degrade and interactions are typically diffusively coupled. At each level [there is the question of] how information is acquired, stored, transmitted, replicated, transformed and robustly encoded. [Krakauer home page, Santa Fe Institute]

Holland (1995) provides a set of properties for a self-organizing system; below is a Wikipedia entry. What distinguishes a CAS from a pure multi-agent system (MAS) is the focus on top-level properties and features like self-similarity, complexity, emergence and self-organization. A MAS is defined as a system composed of multiple interacting agents; whereas in CAS, the agents as well as the system are adaptive and the system is self-similar. A CAS is a complex, self-similar collectivity of interacting, adaptive agents. Complex Adaptive Systems are characterized by a high degree of adaptive capacity, giving them resilience in the face of perturbation.

Other important properties are adaptation (or homeostasis), communication, cooperation, specialization, spatial and temporal organization, and reproduction. They can be found on all levels: cells specialize, adapt and reproduce themselves just like larger organisms do. Communication and cooperation take place on all levels, from the agent to the system level. (Wikipedia Complex Adaptive System page)

Time scales for evolution are also important (Wilkins). On a shorter time scale we are looking at the robustness and plasticity of the system, its responsiveness to stimuli, and to facing new challenges emerging in the environment. Short term provides survival in current and emerging conditions. Longer time scales bring concerns with evolvability, extensibility and repurposability. Long term, core properties make a system continuously adapted to new conditions where major changes may have taken place.

Epidemiologist Stephanie Forrest (2004) has suggested as an example, that secure networks be comprised of diverse operating systems such that the entire network of devices can’t be taken out at
once. There is not one outward-facing firewall but many internal firewalls, possibly down to the chip level. With the outward facing firewall, once it is breached everything is compromised. With built in genetic firewalls as it were, not every device or area can be affected at once. Other constructs relevant to our field are co-adaptation vs. conflict as survival strategies dependent upon scale. This, and other constructs require us to place technological change firmly in a social and even planetary framework, as an actor in the system. There has been some debate on optimal module size (Wilkins, 2014). This would be a debate in information systems as well. “Intermediate” modules with not too many interface boundaries, but not so large as to fail catastrophically are likely optimal. Complex systems tend to be failure resistant and regenerative.

At the scale of local interactions, systems are highly robust because there is no single failure point. There can be considerable breakdown without causing a system failure. Diversity in systems allows even a failed system to restart from a few agents. Take a forest system with birds, insects, trees, bushes, ground plants, soil, water, and temperature. A forested or burnt area can restart with seeds from adjacent forest plants or deposited by birds or animals, insects for birds to eat, new shade produced by the grown plants, more undergrowth, more vertebrates and their fertilizer, more insects, more plants. Modularity and diversity provide not only sustainability but also regeneration in the case of failure.

SOME ANALOGIES

While technological systems are designed and therefore not inherently guided by biological principles, there are parallels in how nature handles the kinds of problems we face as designers of systems. In complexity, initial conditions can be indeterminate, so an intangible such as a purpose, a meaning or an aesthetic can be the basis for a system. Competing forces, different affordances, standardizations, existing constraints in the form of laws and practices, all guide what inventions make it to an innovation similar to what happens in a natural ecosystem. Human inventors scan for opportunities, usually based on existing affordances, but not always. Apple’s platform created an ecosystem for apps that in turn fostered its competitiveness in the device market, while staying true to design standards. Tesla has been able to ignore existing constraints by creating a market, an ecosystem of sales outlets, and coevolved inventions in storage and supply.

As for developers of systems, Bolici et al. (2016) have used biological analogies for cooperation in software development. Their finding was that developers practice stigmergy, a concept from swarm theory that describes cooperation in insect colonies. Ants leave pheromone traces not only of their presence but also of their activities. Ants interact with other ants in real time, but also with the work-in-progress of ants no longer present. Applying stigmergy to software development, the authors studied how developers picked up code in progress and worked directly with the traces of prior effort, without the need to interact explicitly. Social theories that include complexity properties include actor-networks and assemblage theory. Both rely on diversity and agency, recognize the primacy of interaction, are founded on network structures, and have emergence and nonlinearity as dynamics.

CONCLUSION

In summary, this short list of properties suggests several concepts that could be developed for the future of technical systems to be designed for adaptiveness, to be self-reproducing yet variable, and to have a built-in capability to fit to future scenarios. This scope is both broad and challenging, but exploring it promises to generate decades of provocative WG 8.6 conferences. The idea is for a generative mission to accommodate a variety of studies under a common framework such as exists now in the sciences,
including compatible social theories. This is an opportunity to join a powerful scientific revolution and to develop evolvable systems that fit with societal goals. A recent MISQ Call for Papers on Emerging Digital World signals the timeliness of this approach (McElvey et al 2016).

BIBLIOGRAPHY


Krakauer, David Santa Fe Institute statement of purpose. https://www.santafe.edu/people/profile/david-krakauer

Kurzweil, Ray https://en.wikipedia.org/wiki/Mind_uploading


