# 43

# Interaction Is the Future of Computing

Michel Beaudouin-Lafon
Université Paris-Sud, Orsay, France

**P. Wegner, 1997: "Why Interaction Is More Powerful Than Algorithms"**

In 1997, Peter Wegner published an article entitled "Why Interaction Is More Powerful Than Algorithms" (Wegner 1997). The word "interaction" immediately attracted my attention: I knew Wegner's work on object-oriented programming and I was curious to see what he had to say about human–computer interaction (HCI). I soon discovered that the article was not about HCI but about interaction in general, mainly among machines. Nonetheless I was struck by its relevance to my research since I have always tried to understand why programming (i.e., writing algorithms) is so similar to and yet so different from interacting with a computer. Both allow us to make the computer do something, yet they seem irreducibly different from each other.

Wegner's article is both profound and radical. Through a rare combination of theoretical, practical, and philosophical arguments, it challenges Church's thesis, the cornerstone of computer science, which states that everything a computer can do is reducible to what can be done by a finite-state Turing machine. Wegner shows that Church's thesis may hold in the closed world of an algorithm that reads its input, shuts down from the outside world while it computes its result, and then spits it out. But it cannot hold in an open system that harnesses the power of its environment by interacting with it as it runs, because the environment provides an endless stream of unpredictable events that cannot be reduced to an algorithm. This has wide implications in many areas of computer science (Goldin, Smolka, and Wegner 2006) and has attracted sharp criticism from theoretical computer scientists (few dare challenge Church's thesis), but I want to focus here on Wegner's philosophical argument and its impact on my own research in HCI.

Most modern sciences, including computer science, are strongly influenced by the rationalist and positivist belief that the workings of Nature can be completely captured by mathematical models that allow us to perfectly predict and control them. In computer science, for example, software engineering relies on formal methods to prove the correctness of large software systems; the semantic web assumes that human knowledge can be captured unequivocally by XML descriptions; and the goal of artificial intelligence is to reproduce human behavior with algorithms inside a computer.

Despite being born in the country of Descartes (or perhaps because of it), I have always thought that rationalism conveniently ignores real problems by concentrating on what can be fully understood and controlled. Indeed, all three examples above fail when confronted with the real world: formal proofs of software systems that interact with their environment are impossible unless one makes strong assumptions about the environment, typically reducing it to an algorithmic behavior; the semantic web finds it intractable to define ontologies and map them to each other when they change all the time; and artificial intelligence focuses on mental processes such as problem solving but usually ignores their tight coupling with human perception and action.

While the so-called hard sciences such as theoretical physics deal with the basic phenomena of matter and energy, the oft-despised "soft" sciences such as psychology and sociology deal with far more complex and subtle systems that clearly cannot be described by pure logic and equations. As Wegner explains, empiricists sacrifice the rationalist completeness and predictability of closed-world systems in order to address open, interactive systems in the real world. My own interpretation of this is the *myth of perfection*: under the rationalist assumption, complete and perfect control of a process is always possible; it is simply a matter of getting the model right. Wegner's notion of interaction shows that this is impossible in general, and that debunking the myth of perfection brings more power, not less.

This argument actually helped me get respect from my computer science colleagues. As a member of a hardcore computer science department, I have often felt that my work in HCI (or rather, "user interfaces," as they call it) was considered as just painting pretty pixels on the screen and giving cool demos. Telling them that interaction (what I do) is more powerful than algorithms (what they do) not only triggered interesting and controversial discussions, it also helped me analyze some of the evolutions in computer science and HCI.

Empirical approaches and interaction, in Wegner's sense, are indeed slowly becoming more common in computer science. For example, distributed systems are now ubiquitous, from the Internet to computer clusters and multicore chips. Such large and complex systems can no longer be analyzed as a single algorithm but must be seen as a set of interacting entities. Probabilistic approaches, approximate algorithms, and stochastic methods are also more widely used to deal with uncertainty, incompleteness, or simply the fact that it is better to compute a good solution in a short time rather than to wait forever for the perfect one. The latter is a feature of so-called anytime algorithms, which provide the best possible solution at any time in their execution, improving it if allowed to run longer. Such algorithms can be used to create mixed-initiative interactive systems where humans and computers take advantage of each other's expertise by running in parallel and interrupting each other (Scott, Lesh, and Klau 2002), a perfect illustration of Wegner's notion of interaction.

In HCI, the rationalist approach is less predominant but still very strong. Much research is based on the belief that we can capture human behavior formally, for example, by some task model, or that the results of controlled experiments can be taken as objective measures of the phenomena being tested. Reducing human behavior to such models is indeed tempting: it turns interactive system design into the relatively simple problem of defining the set of widgets needed to accomplish specified tasks.

I find it more interesting to design an interactive system without making unnecessary assumptions about how it will be used. I believe that an interactive system should be like a canvas for a painter, a medium to express oneself whose power comes from the freedom it gives rather than the constraints it imposes. HCI has a long tradition of considering interactive systems to be open to (re-)interpretation by their users. Informed by ethnographic work that repeatedly demonstrates that humans do not always behave in rational and predictable ways (see, e.g., Suchman 1987a), it includes such empirical approaches as participatory design (Greenbaum and Kyng 1991), end-user development (Lieberman et al. 2005) and coadaptation (Mackay 1990).

The instrumental interaction model I created (Beaudouin-Lafon 2000) stems from this same body of work, although its purpose was more operational. Instrumental interaction was inspired by the observation that humans create tools and instruments to empower themselves, to do things that they could not otherwise do, whether hammering in a nail, playing music on a piano, or putting together a budget with a spreadsheet. Instruments allow us to harness the power of the environment (here, the computer), exactly as advocated by Wegner. In fact, Wegner's paper was instrumental in making me focus on interaction rather than on the interface itself (Beaudouin-Lafon 2004), that is, on the mediation between users and computers and the capture of this mediation into interaction models.

Wegner defines interfaces as behavior specifications. This is sufficient when focusing on machine-to-machine interaction because the interaction is symmetrical: the interacting entities are similar in nature. In contrast, human–computer systems exhibit a stark asymmetry between the human means of communication and the computer's input and output devices. To resolve this asymmetry and to mediate the interaction requires the *reification* of the interaction, that is, the creation of a new object, the instrument, that translates between the languages of the two parties. Wendy Mackay and I developed this notion of reification together with other design principles in order to operationalize further the design process of interactive systems (Beaudouin-Lafon and Mackay 2000). Our goal was, and still is, to move from descriptive models of interaction to generative ones, not in the sense of automatically generating interfaces from abstract descriptions but instead by providing tools for designers to both expand and channel their creativity. Such *generative theories* are, in a sense, a tribute to Wegner's plea for empiricism over rationalism, for interaction over algorithms.

Finally, Wegner's conception of interfaces as harnesses resonates with my work on interaction techniques. Since people rely on their perceptual and motor skills to interact with computers, we need to explore how best to optimize these skills to harness the power of the computer, and vice versa. For example, my joint work with Yves Guiard on multiscale pointing and navigation (Guiard and Beaudouin-Lafon 2004) shows that Fitts's law still applies to very high indices of difficulty, that is, for pointing tasks that are inaccessible in the physical world, a clear demonstration of Wegner's concept of interface as harness.

In conclusion, I consider Wegner's article a landmark in computer science, a work that opens a window onto a new world with large areas yet to be explored. This work has often been misunderstood or dismissed, as if the new light it shed was too bright to discern anything clearly and it seemed safer after all to close the shutter. I believe that the goal of HCI is not to make pure computations somewhat more palatable for human consumption, but instead to redefine the role of information and computers in our ecology, that is, to create a paradigm shift (Kuhn 1962) from computation to interaction.

Returning to my original question about the respective natures of interaction and programming, one may draw a distinction in terms of scale. Algorithms are but building blocks in larger interactive systems, like drops of computation in a sea of interaction. Rather than trying to understand waves and currents by observing drops under a microscope, Wegner looks at the whole system on a larger scale. As with other natural and artificial systems, complexity arises from emerging behaviors and the effects of effects through a slow evolutionary process. By focusing less on algorithms and more on interaction, computing is starting to grow out of its infancy. Interaction is its future.

### *References*


Beaudouin-Lafon, M. 2000. Instrumental Interaction: An Interaction Model for Designing Post-WIMP User Interfaces. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 446–453. New York: ACM Press.

Beaudouin-Lafon, M. 2004. Designing Interaction, not Interfaces. In Proceedings of the Working Conference on Advanced Visual Interfaces, 15–22. New York: ACM Press.

Beaudouin-Lafon, M., and Mackay, W. E. 2000. Reifi cation, Polymorphism, and Reuse: Three Principles for Designing Visual Interfaces. In Proceedings of the Working Conference on Advanced Visual Interfaces, 102–109. New York: ACM Press.

Goldin, D. Q., Smolka, S. A., and Wegner, P. 2006. Interactive Computation: the New Paradigm. Berlin: Springer-Verlag.

Greenbaum, J., and Kyng, M. 1991. Design at Work: Cooperative Design of Computer Systems. Mahwah: Lawrence Erlbaum.

Guiard, Y., and Beaudouin-Lafon, M. 2004. Target Acquisition in Multiscale Electronic Worlds. International Journal of Human Computer Studies, 61 (6), 875–905.

Kuhn, T. S. 1962. The Structure of Scientifi c Revolutions. Chicago: University of Chicago Press.

Lieberman, H. 2001. Your Wish Is My Command: Programming by Example. San Francisco: Morgan Kaufmann.

Mackay, W. E. 1990. Users and Customizable Software: A Co-Adaptive Phenomenon. Ph.D. thesis, Massachusetts Instititute of Technology.

Scott, S. D., Lesh, N., and Klau, G. W. 2002. Investigating Human-Computer Optimization. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, 155–162. New York: ACM Press.

Suchman, L. 1987a. Plans and Situated Actions: The Problem of Human–Machine Interaction. Cambridge: Cambridge University Press.

Wegner, P. 1997. Why Interaction Is More Powerful Than Algorithms. Communications of the ACM, 40 (5), 80–91.