Cybernetics and Human Knowing. Vol. 18, nos. 1-2, pp. 153-175

Information and Semiotic Processes

The Semiotics of Computation

Mihai Nadin[1]

Information processes and semiotic processes are complementary. While some aspects of physical reality can be convincingly described in terms of information—understood as an expression of entropy—the living conjures descriptions that integrate the meaning of change. Semiotics is the domain knowledge of representation and interpretation. It is against this conceptual framework that we discuss attempts at involving semiotics in the analysis and design of information systems, of human-computer interactions (HCI), and of programming languages.

Keywords: anticipation, grounding, information systems, representation, semiosis

Introduction

The subject of this article is easy to describe: Semioticians argue that their knowledge domain is relevant to computer science. But if so, then why do computer scientists, with very few exceptions, continue to ignore semiotics? This question can be reformulated: Can semiotics make a difference in our understanding and practice of computation?

The fact that mathematics is relevant to digital processing needs no proof. Computers are automated mathematics at work. Logic (at least Boolean logic) is just as relevant. The fact that physics is essential in conceiving and making computers is also undisputed. The same holds true in respect to chemistry: Consider the processes through which chips are made, VLSI (very large scale integrations) are produced, and various components (memory, I/O devices, etc.) are engineered. Furthermore, people involved in computer science will acknowledge inspiration from biological models, psychology, cognitive science, ergonomics, ethnographic studies, sociology, graphic design, and product design. Communication theory, which some will identify with Shannon and Weaver's *Mathematical Theory of Communication* (1949), is also considered as an inspiration. Some computer scientists acknowledge interest in language-based theories (especially formal languages) concerning what is needed to successfully "communicate" with a machine. But semiotics? If it ever shows up on the radar, it is rather a blip than a foundation of computer science—contrary to what many semioticians have claimed or hoped is the case.

There are aspects of computation in which semiotics has fared relatively better: in particular, human-computer interactions (HCI). But even within HCI, there is no real consensus regarding the practical relevance of semiotic considerations or the possibility to apply them. In a short article in *Communications of the ACM*, Zemanek

1. antÉ – Institute for Research in Anticipatory Systems, University of Texas at Dallas. Hanse Institute for Advanced Studies, Delmenhorst, Germany. Email: nadin@utdallas.edu

(1966) suggested that semiotic analysis is relevant in analyzing programming languages—to no avail. Even those (like Peter Bøgh Andersen, Berit Holmqvist, Jens F. Jensen, Ronald Stamper, Kecheng Liu) who eventually ventured into information systems, were not able to produce more than some considerations, using semiotic terminology—usually not strictly defined—of no real practical relevance. Andersen (1990) published the first book on the subject and taught himself programming in order to prove some of his ideas. In *Mind at Work*, he wrote, "If a system is to count as a theory, its program text must be interpretable as the kind of statements that make up a theory," (Andersen, 2003, p. 9). He was referring to Nadin's (1982) attempt to utilize semiotic theory in a broader epistemological perspective. While at the University of Aalborg, he maintained a bibliography on semiotics and informatics, which I wish someone would continue.

Semioticians discuss symbol processing, while others, mainly in the artificial intelligence (AI) domains practice it. If this judgment sounds too drastic, it is because it describes a state of affairs that, self-delusional statements to the contrary, so far has failed to change. Although this review is focused on particular books—*Semiotics in Information Systems Engineering* by Kecheng Liu (2005); *The Semiotic Engineering of Human-Computer Interaction* by Clarisse Sieckenius de Souza (2005); and *The Semiotics of Programming* by Kumiko Tanaka-Ishii (2010)—we are interested in a broader perspective. The question formulated at the beginning of this article—Is semiotics of any use for those working in computation?—extends to the authors and books just mentioned, but is not limited to their particular contributions.

Shannon Understood Semiotics Better Than We Do

The typewriter on which Shannon (or his secretary) typed (July 7, 1953) his thoughts on non-numerical computation had a very simple interface. Eventually, after many years of having computers driven by punch cards, IBM took the mechanical typewriters that the company was successfully producing and put them (literally!) on top of their computers. In the given context, this was a semiotic decision. It is always easier to continue something (accumulated knowledge and experience) than to force new means and methods upon human beings. (This corresponds to the mimetic phase of new technologies.) Years later, Douglas Englebart came up with a *mouse*—another semiotic decision, translating coordinates in real space into the space representation on a computer monitor. This time, the pointing device introduced a new vocabulary of actions. An interrupt procedure, not unlike the electric switch in our homes, changed the nature of commands: from text-based to image-based, from one language (commands in a natural language) to another (visual). It also started changing our cognitive condition. So much more happened afterwards. Before you knew it, the mouse got new functions and became wireless. And so much is taking place in our days: gestures and expressions drive all kinds of programs. An undergraduate student is trying to join my research group, stating interest and acquired competence in affective interfaces, adaptive and intelligent interactions of machines, augmented reality navigation. The human mind shaped in these experiences is different from that of the human beings who used to write texts on paper, or from those working in the programmed assembly line (the Taylor method for achieving efficiency and consistency of performance). Information rich environments can no longer be avoided.

Shannon did not foresee the particular technical developments described above. His text (later re-titled *Computers and Automata*) makes reference to Samuel Butler's *Erewhon* (a most engaging social satire), originally entitled "Darwin Among the Machines." In Shannon's formulation, "In the topsy-turvy logic of satirical writing, Butler sees machines as gradually evolving into higher forms" (Shannon, 1953 p. 1234). He further observes, "Current and projected computers and control systems are indeed assuming more and more capacities and functions of animals and man"(Shannon, p. 1234). A powerful prediction follows: "the most exciting potentialities of computers lie in their ability to perform non-numerical operations" (p. 1234). This means, "logic machines, game-playing machines and learning machines" (p. 1234). This was articulated almost 60 years ago. While today's prophets of the singularity (Vernor Vinge and Ray Kurzweil) stir our imagination with a model of machines that have abilities higher than those of the humans who conceived them (super-intelligence), Shannon's analysis points to computation as augmenting our abilities. (As visionary as he was, he did not foresee the changing of our abilities.) This might sound as old-fashioned and lacking the romantic edge of science fiction. In order to make his argument, he dealt with details, showing how the mechanical switch technology he had to use could support playing checkers, learning, and self-reproduction functions. Most important: Such performance was an expression of complexity—a term that "singularites" have yet to learn how to spell (never mind to understand). Shannon was enthusiastic about machines; he did not exclude the possibility of having a robot elected president of the USA. That he missed a subject such as the wedding between a human being and a robot testifies to values characteristic of his time, not to lack of imagination.

So far, no explicit semiotics. The implicit expectation is that the living (Shannon's reference is to animals and man) can have "more capacities and functions"(Shannon, 1953, p. 1234) than the rest of the world; moreover, that within the living, various phenomena don't only take place, they also have meaning, and the meaning partakes in their identity. For those even marginally informed about Shannon's communication model, the explicit stipulation that transmission of information—which was his assignment at Bell Labs—is independent of its meaning, but dependent on the properties of the channel, is symptomatic of his awareness of the distinction between the informational and the semiotic. He focused on information. "A difference makes a difference" (Gregory Bateson's description, *Steps to an Ecology of Mind*, 1972); information meant a measure of the reduction of uncertainty (entropy). In Shannon's model, information is bits per second (or bytes, or kilobytes, etc. per time unit), a commodity subject to the logistics of very complicated networks (copper wire, fiber, wireless, etc.)—but unrelated to meaning. The bit describes the degree by which uncertainty is decreased when something with a 50% probability happens (the classic heads-or-tails experiment with a coin). An event that has a 25% probability (one-fourth) yields two bits of information. This is why Shannon, after delivering to the scientific community methods for encoding messages in zeros and ones (pretty much like Leibniz did centuries before him), focused on symbolic computation. He eventually became a contributing member of the "gang" (together with John McCarthy, Marvin Minsky, Nathaniel Rochester) that initiated the Dartmouth Conference—which launched artificial intelligence as a distinct knowledge domain. They were semiotically driven, that is, looking at language use to form new abstractions and concepts. Shannon's fascination with toys and games (he wrote an early chess program, see Shannon, 1993) took over the dry scientific work and made him into a *homo ludens* (playful human), obviously a particular form of what Felix Hausdorff (1897, writing as Paul Mongré) would call *zoon semiotikon* (semiotic animal).

The composite image of Shannon's activity makes one thing clear: Information processes, for which he provided a mathematical foundation, are complementary to semiotic processes. To use the metaphor of the coin, so often associated with Ferdinand de Saussure's (1968) semiology: One side is that of information, that is, quantified data; the other, semiotics, of qualitative distinction brought about through interpretation. Together they form a coherent unity, exemplified by, among other things, what is called computation. (Saussure made the analogy to a sheet of paper; see. p. 254.) The red-light system is a very good example of the relation between information and meaning: Green, yellow, red have a meaning defined in the law (traffic regulation). They are also subject to switching, and therefore we can attach an informational value (while driving we can expect the green light to switch to red). The number of bits corresponding to the rhythm of switching is not related to the meaning of the three values. But intelligent red lights could provide optimal switching (to maintain the flow of traffic without ignoring the rights of pedestrians).

To make this even more clear: Shannon's chess program was a beginning. It encoded information—the rules, the roles—and meaning (What does it means to endanger a knight? What is a draw? A win?) Botvinik, the world chess champion at the time, beat Shannon handily. Today, computer programs can defeat the best players. The outcome, however, is a bit disappointing: the games are not exciting; they are precise. The programs perform in the informational space—vast computing resources that can be deployed in calculating many moves in advance. Plus: The computer chess player can consult a huge library of games. This library documents real situations, encountered over time by large numbers of players. The stored data can be used in meeting any challenge. The semiotic space is secondary. But chess is more; better yet, it involves not only information, but also meaning, creativity. The game encodes culture: a society organized hierarchically, conflicts carried out according to rigid rules. The cultural dimension, expressed in the beauty of the game is lost when chess is reduced to information processing. The same applies to creativity. Regardless of the domain of expression: music, multimedia, theatre, dancing, poetry, and so forth, it involves information, but also conjures meaning. It invites to interpretation. Otherwise it makes little sense. Splashing colors on a canvas does not make everyone a Jackson Pollock! There is meaning to action painting—those images are testimony to feelings and emotions, and not to how things look when we stare at them or take pictures of them. The fact that some modern artworks look like any child can make them does not mean that every child can make them as meaningful images, that is, as an expression of the intention to share something. Aesthetic awareness, like semiotic awareness, qualifies the action as significant, or insignificant (as it often is the case).

What is the Knowledge Domain of Semiotics?

Liu studied computer science and information management. He is convinced that "the word *semiotics* comes from the Greek for symptom," (Liu, 2005, p. 13). After several more dubious statements (e.g., "Ferdinand de Saussure … founded semiology, a European school of semiotics"), he writes: "There are three distinct fields of semiotics: syntactics (or syntax), semantics and pragmatics," (Liu, p. 13). There is no more semiotics, or better semiotics, in the rest of his book. Over the last two years, I have been in touch with some of Liu's students. One of them explained to me that, "His interest is in applying organizational semiotics methods in system analysis and design"(personal communication, April 21, 2011).

Sieckenius de Souza and Tanaka-Ishii have a background in linguistics. Their take on semiotics is more precise. Sieckenius de Souza writes about "the study of signs, signification processes, and how signs and signification take part in communication" (Sieckenius de Souza, 2005, p. 3). Tanaka-Ishii states that, "The most fundamental semiotic question … is that of the basic unit of signs" (Tanaka-Ishii, 2010, p. 26). Liu is focused on information systems. Ronald Stamper (Twente, Netherlands) initiated early on (in 1973) what today we call IT-based systems, and in 1989, Liu joined his team. Sieckenius de Souza is focused on human-computer interaction; Tanaka-Ishii on programming languages. Were it not for the word *semiotics*—at times used in a more than approximate manner—these three books would have nothing in common. (That two of them bear the imprint of Cambridge University Press means nothing.) The fact that the computer is at the center of their inquiries is also marginally relevant. Not one of them was interested in defining computation from a semiotic perspective. Since this was not their explicit goal, to reproach (not to say criticize) the absence of such a subject cannot be justified. The current dominant perspective of computation is information processing. However, computer scientists realize that in accepting this reduction, they miss opportunities associated with what they call the *symbolic dimension* (see Conery, 2010). This means interpretation and results from the need to deal with meaningful information.

But is it possible to address semiotic aspects of computation without trying to define the broader semiotic perspective? Moreover, while information theory needs no defining of its knowledge domain—this is exact science—can we assume that this is also the case with semiotics? (Some scientists will argue that information itself is not defined in a universally accepted manner.) The three authors rely on three totally different understandings of what semiotics might be. In this respect, they are representative of the entire discipline. Indeed, for those reading articles or books on various aspects of semiotics, it is striking that the underlying understanding of what semiotics is does not exist. Maybe the assumption that semiotics deals with signs can be conjured as a commonality (or banality). But this does not help either, because there is no consensus on what this entity called *sign* is. Most of the time, previous definitions are carried over, but not in the precise manner that a soloist would read a score. Rather, these definitions take the shape of memories, of a time reconstructed from renditions originating with musicians who do not know how to read a score, or don't care to read one. In order to understand the unity between information and semiotics, that is, between processing data and acknowledging meaning (as a prerequisite for practical activities), the researchers ought to define the *relata*, that is, the entities brought into relation. Famous as a physicist, John Archibald Wheeler insisted on the meaning of information (see Davies, 2004, pp. 8-10). A bit has to refer to something and we need to understand the reference. A click of the Geiger counter is connected to knowledge (his example: The atom has decayed).

Those seeking an understanding of semiotics in connection to practical activities rely on a large body of shared knowledge. In particular, the experience accumulated in various forms of interaction among individuals and within communities is such a source of knowledge. Interactions between the human being and the rest of the world are also relevant in defining the knowledge domain of semiotics. Epistemology reflects the effort of finding traces of initial questions in activities. We have no access to the brains of those who early on transcended the immediateness of their actions and looked for the "beyond"—further in the space of their existence, further in the time of their own lives. But we do have access to the processes through which newborns and infants (humans and animals) acquire experience with representations—and how they produce their own as needs or circumstances require. And we know (see Mitchell, 2009) that the human brain is a living testimony to the interaction of minds: the three most developed active brain regions are specifically in service of understanding the "goings-on of other people's minds"(Mitchell, p. 79). This understanding is not about the chemistry or electricity of the brain, rather about the meaning of human actions, sense of future. In short, it is about the semiotics based upon which individuals aggregate in order to achieve goals otherwise not within their reach. Semiotics is, in this context, defined through the focus on representation (not signs), more precisely, capability of "representing mental states in others" (Gallese, 2001, p. 33).

Knowledge, in its simplest forms, originates in the awareness that there is something beyond immediacy. This applies to physics—How does the world behave?—as it does to mathematics, to logic, and to semiotics. Geometry originates in activities related to sharing space. Such activities can be: laying claim to portions of the surroundings; exercising ownership; initiating or participating in exchange; production; market processes, and so forth. The variety of forms through which geometry participates in such activities is testimony to its "making"on the fly, as needed. There are no points, lines, or surfaces in the world—these are late abstractions to which computer-based geometry laid claim. There are no numbers in the world, although there are mathematicians (Livio [2003] is one of them) still convinced that numbers exist just as stones and plants exist. Just as there are semioticians who "read" the "sign of nature"—or other signs, some not even worth mentioning—or believe that what they call signs exist in reality, regardless of whether we interpret them as such or not. To measure a surface, that is, to introduce a scale, is related to practical tasks. These become more creative as more means for qualifying the characteristics of an area are conceived and deployed. To measure is to facilitate the substitution of the real—the measured entity—with the measurement, that is, the representation of what is measured. To travel, to orient oneself, and to navigate are all children of geometry, in its intercourse with semiotics. Extended from the immediacy of one's place to its representation, geometry and semiotics fuse. The experience of watching stars and observing repetitive patterns in the environment translate into constructs, which are integrated in patterns of activity. Rosen (1985, p. 155) took note of "shepherds [who] idly trace out a scorpion in the stars" (the subject of interest being relations among components). He also brought up the issue of observation: "Early man … could see the rotation of the Earth every evening just by watching the sky" (Rosen, p. 201). Rosen's suggestion, in the spirit of Hausdorff's definition of the semiotic animal (to which we shall return), is that observations, which vary from person to person, do not lead to uniform inferences, are not automatic: An early observer "could not understand what he was seeing," as "we have been unable to understand what every organism is telling us," (Rosen, p. 201). Because we did not have the knowledge for it. Based on these rather common sense thoughts, we can attempt a first observation: The "language" in which phenomena (astronomic or biological, or any other) "talk" to us is that of semiotics. Let us generalize this observation: Our entire knowledge, from the most concrete to the most abstract, is embodied in the variety of languages that define the sciences. Each scientific language encapsulates the raison d'être of the respective knowledge domain. Mathematics, in its more comprehensive condition as an expression of abstract knowledge, is a view of the world as it changes. Computer science is a view of the world resulting from the hypothesis that everything is reducible to information processing (life itself being a computation, part of the comprehensive computation that makes up the universe). Von Neumann (1963) was by no means the only one to voice this viewpoint. The semiotics of life is quite often incorporated in the processing of information. The meta-level of knowledge is reabsorbed into the specific domain knowledge.

Within the scope of this article, it is, of course, not necessary to re-write the history of semiotics. Nevertheless, those aware of its history—the three authors whose books are discussed would not claim even a stake in this history—know that defining the knowledge domain of semiotics and, moreover, identifying its specific methods, is unavoidable. In the absence of knowing what and why we research something, moreover which methods we utilize, there is no knowledge to be accounted for.

Semiotics began and failed many times. Plato's dialog *Cratylus* (360 BCE/2010) brought up the laws (*nomothétes*) that semiotics is expected to describe (actually, naming is the action, but it corresponds to naming causes). Not that semiotics itself is an expression of law, rather a means for evincing it. We see associated to names the force (*dynamis*), and we read about relating things to names. Of essence is the discourse—which is how semiotics was embodied at that time. Aristotle's *Poetica* (350 BCE/1961), the contributions of the Stoics, of Sextus Empiricus (*Adversus Mathematicos*, *Commentaries on the Stoics*, VIII) about 450 years later—all remain sketches, collections of thoughts worth our intellectual effort to understand, but of no consequence to those involved in the study of computation. The fact that Brenda Laurel, a Ph.D. candidate at the time, who I met during my computational adventure at The Ohio State University, used in her thesis Aristotelian mimesis to address issues of human-computer interaction contradicts my assertion—at least formally. (The massively multi-player game called *Mimesis* has nothing to do with the subject.) In reality, neither the signified (*lekton*), nor St. Augustine's (397/1958) *De doctrina cristiana*, nor St. Anselm's Monologion (1075-1076, see Williams, 2007), nor the Arabian explorations (Avicenna, in particular), and not even Aristotle's *mimesis* could help us better comprehend what it means to compute. Yes, their concepts (I named only a few) are traces of questions posed early on, in particular: How can something in the world in which actions assure our performance (survival) be "duplicated" in the mind? Lambert's questions (1764) concerning the connection between thinking and things belong to the same category. Awareness of the never-ending reciprocal questioning of minds comes up relatively late (see Nadin, 1991, in particular the origin of the English word *mind*).

To be very clear: It is not the early or the more recent history that is of particular relevance in this respect, rather the attempt to understand the need for semiotics—if such a need indeed exists. None of the books under discussion here—and for that matter, very few of the semiotic contributions to the subject of computation—testify to such a need. The Middle Ages—yet another beginning of semiotics (Roscelin, Guillaume de Champeaux, Garland, Abélard)—does not change the situation. Who, if anyone, would find in Jean de Salisbury (*Metalogicon*) arguments for the relevance of semiotics? At best, we find traces of yet other questions—such as those referring to the nature of abstractions—in the reflections of Occam, William of Shyreswood. Lambert d'Auxerre, and Roger Bacon regarding what is needed to reach clarity. No doubt, Hobbes (Leviathan, 1650/2010), the *Logique de Port Royal* (Arnaud & Nicole, 1662/1964), and John Locke (forms of reasoning and, more important, *The Division of the Sciences,* 1690/1979) are precursors of the modern rebirth of semiotics, associated with Ferdinand de Saussure and Charles Sanders Peirce. More important, especially in view of our focus on computation, is Leibniz' vast opus. There is no semiotics as such in Leibniz, but there are the *lingua adamica* (a universal language into which everything can be translated) and the *calculus ratiocinator*, which was probably way ahead of the computing machine he owned (and which some believe he built). As different as the views of Leibniz and Peirce are, this is the closest we come to realizing why semiotics could be significant for understanding computation or, for that matter, the workings of the human mind.

Important, even for those disinclined to seek guidance in works of the past, is the distinction between language associated with convention or law (*nomoi*) or with nature (*phusei*). Nobody expects today's semioticians to become historians. But in the absence of a broader understanding of our concepts, we will continue to explore, blindfolded, new continents (of thought and action). I do not doubt that Saussure and Peirce are valid references, but I suggest that Hermann Paul's (1880) *diachrony* is far more conducive to understanding the specific dynamics of languages. This is only one example. Nikolai Sergeyevitch Troubetzkoy might be another, as is Louis Hjelmslev.

Computers before the computer

If mathematics, or logic (or both) is a universal language, is a machine conceivable for automating the practical activity characteristic of mathematics? Before the machine, there were, of course, human computers:

Figure 1: What a "Computer" Was Supposed to Know

A COMPUTER WANTED.

WASHINGTON, May 1.—A civil service examination will be held May 18 in Washington, and, if necessary, in other cities, to secure eligibles for the position of computer in the Nautical Almanac Office, where two vacancies exist—one at $1,000, the other at $1,400.

The examination will include the subjects of algebra, geometry, trigonometry, and astronomy. Application blanks may be obtained of the United States Civil Service Commission.

The New York Times
Published: May 2, 1892

Note: The human being as a computer predates the automation of mathematics.

Leibniz and Peirce were, for their respective times, good mathematicians. Thus they understood what was necessary for being a good computer—human or machine. Among those prerequisites are:

- understanding representation—to operate on real objects is different from operating on representations;
- understanding the meaning of operations on representations;
- the ability to evaluate the outcome of actions, that is, the performance.

Peirce contributed a text entitled “Logical Machines” to the first issue of *The American Journal of Psychology* (November 1887, pp. 165-169). He made reference to “Voyage to Laputa” (from *Gulliver's Travels*), in particular to a machine for evolving science automatically. Jonathan Swift’s ironic take on Aristotle and Bacon is contrasted to the logical machines of Jevons and Marquand (who was one of Peirce’s students), and to the mathematical engines (Webb’s adder, Babbage’s analytical engine) of his own time. The subject, however, is “the nature of the reasoning process” (Peirce, p. 165). Those willing to take time will enjoy the detailed presentation of the logic at work in such machines. Readers in a hurry (and this is the rule in our time) will notice some telling observations:

- Every machine is a reasoning machine.
- Experiments make evident “the objective reason embodied in the laws of nature.”
- Every reasoning machine … has two inherent impotencies:
    1. destitute of all originality, of all initiative, it cannot find its own problems;
    2. it can perform only what it was conceived for (Peirce, 1887, pp. 168-169).

Take note: There is no suggestion that semiotics might be of any relevance to understanding what the machines Peirce described in his article are. We should probably understand that the knowledge domain of semiotics does not include any machine, neither the abacus nor the most recent embodiment of digital or analog computation, rather, what machines might *process*, *compute*, if information and meaning together could become the object of their operation. In addition, they would have to reunite deterministic and non-deterministic aspects characteristic of cognitive processes. Reciprocally, the knowledge domain of computation includes semiotics, whether implicitly or explicitly, because regardless of the nature of computation, it requires representations and interpretations, and assumes interactions mediated by representations with living entities. Indeed, machines operate on real entities when they process them (polishing diamonds, making flour out of wheat, making chips, etc.). Computation operates on the re-presentation of the real even when they drive processing machines. Regardless of their degree of sophistication, the output of semiotic processes, known also as *semioses*, is always subject to a new process of interpretation, to a new semiosis. Please make note of the distinction of the variety of machines.

But Peirce would not be Peirce if he had not revised his ideas from “Logical Machines.” Lauro Frederico Barbosa da Silveira (1993) documents the change of mind and the difficulties researchers have when they refer to C. S. Peirce (which one?). She discusses semiotic machines—a concept based on logical machines—and identifies learning as a defining characteristic of such machines. So much to uncover. So much to understand.

Degree of Necessity

This is a very simple logical procedure: Imagine that semiotics vanished. Given its relative inadequacy of making possible knowledge otherwise not available, it did die many times. I took Liu's book, ignored the lack of terminological discipline, and rewrote it (the miracles of digital processing!), leaving out the semiotic terminology. It turns out that the book remained basically the same. Liu's focus is on information. When the author defines organizational semiotics, that is, the study of organizations using the concepts and methods of semiotics (Liu, p. 19), the expectation is that the concepts and methods be clearly defined. But this is not the case with his own writing. Liu writes about divisions of semiotics (confusing levels of semiotic analysis with branches). His work is informed by that of Stamper (1973). This is how other branches are introduced (Stamper's contributions): physics, ("concerned with the physical aspects of signs at the level of signals and marks" [*sic*, Liu, 2005, p. 26]); empirics ("to study statistical properties of signs when different physical media and devices are used" Liu, p. 26); and the social world ("where the effects of the use of signs in human affairs are studied," Liu, p. 27). If only Occam's razor would be at work as authors (Liu and Stamper are not exceptions) continue to multiply the concepts. They do not realize that a tremendous effort was already made to reduce the variety of semiotic entities to what is needed in order to describe semiotic processes coherently and consistently. Of course, we are all entitled to conceive of our own terminology and to suggest new methods.

There is no expectation of dogmatic alignment that should ever censure a scientist. But researchers are also responsible for their attempts to elucidate simple or complex aspects of reality. Moreover, when a researcher builds upon someone else's work—terminology, in particular—they owe it to them to maintain integrity. Let me suggest that Kecheng Liu would be better off in his work without any semiotics terminology. He gives "An Example of Semiotic Analysis" (how an organization works as an information system). I shall limit the quotation (no, I did not make it up); but if you want to have a good *Erewhon* moment, go for the entire so-called "six semiotic aspects":

> *At the physical level*, the telephone must be connected by the phone line through telephone service providers.
> *At the empirical level*, the voice signals will be converted into electronic (or optical) signals and transmitted between two telephones (Liu, 2005, pp. 35-36).

There is no semiotics to start with. In the chapter entitled "From Semiotic Analysis to Systems Design," the subject is converted to the relationship between semantic models and database design. To dwell more upon this kind of vacuous focus on a (nonexistent) semiotics (or semiotic alibi) would not benefit our broader discussion of the relevance of semiotics. It might well be that Liu is competent in systems engineering—and that those who reviewed his manuscript thought they did semiotics a favor. They did not!

Without any doubt, the books by Sieckenius de Souza and Tanaka-Ishii require the semiotic perspective. At least they make a good case in its favor. Their respective arguments, very different although obviously of linguistic inclination, are convincing. *Semiotic Engineering* is an original contribution, which we can only hope will continue to find its way in the HCI community. After this first book, Sieckenius de Souza published *Semiotic Methods for Scientific Research in HCI* in collaboration with Carla Leitão (2009). Her work resulted in a methodology embraced by the HCI community, and was recognized through the Rigo Prize awarded to her in 2010. The semiotic inspection method (SIM) and the communicability evaluation method (CEM) embody experience in the qualitative aspects of HCI. Many examples convincingly make clear the goals and evaluation procedures used to find to which extent goals are attained. The author makes no contribution to semiotics—that is not the purpose of her research—but to engineering. Image after image, the reader becomes familiarized with the functioning of particular programs and, indirectly, with the semiotic implications of the attempt to improve the communication between the user and the program. Sieckenius de Souza understands that design as such is semiotically based, and pays attention to factors as diverse as aesthetics, psychological, social. Eco's program for semiotics, that is, the investigation of signification and communication, is adopted and followed through. From my own perspective, the notion of infinite semiosis (resulting from Peirce's foundation) that Sieckenius de Souza adopts is very encouraging. Recursivity, as she correctly takes note of, is where machines and semiotics meet (Sieckenius de Souza, 2005, pp. 26-27). In a recent e-mail, she expressed frustrations worth bringing up here: "Having studied semiotics does make a difference … I have the impression…that HCI professionals and students educated in North America tend to have a 'What is in it for me?' approach. … As you know, the answer is, 'a whole new world, but it will take a lot of critical thinking to get it'"(Sieckenius de Souza, personal communication, May 19, 2010).

While her frequently simplified semiotics can be frustrating for semioticians, it is obvious that she makes compromises for the sake of the approach. Too many well-intentioned scholars gave up on semiotics because its language is often obscure, or because it introduces difficult, and usually inconsequential, distinctions. Engineers are not known for their predisposition to theory. They invented (17th century, William Oughtred, using Napier's work on logarithms) the slide rule (slipstick), and later on the pocket calculator and the spreadsheet. That's mathematics reduced to what engineers need for routine work. Many more engineers would adopt semiotic engineering if automated means for its applications could be provided. Something like a semiotic slipstick. But does this hold true also for computer scientists? It should be noted that Sieckenius de Souza's activity became an example for others. Her semiotic engineering group is productive; more young researchers seem dedicated to acquiring scientific knowledge for their particular engineering purposes (e.g., web scripting for improved accessibility, design of new interfaces, multi-cultural applications). At the same time that Sieckenius de Souza received the RIGO Award of the Special Interest

Group in Design of Communication (SIGDOC), it was also presented to Maria Cecilia Calani Barananauskas (Institute for Computation, Universidade Estatual de Campinas, Brazil). She is also involved in HCI, and affiliated with Liu's group in Reading, UK.

Sieckenius de Souza tries to address the connection between the developers of software and those who will eventually use it. Communication is therefore the focus. To ascertain that the computer is a sign capable of generating signs is to miss the necessary constructivist nature of semiotics. The fact that Peter Andersen and, especially, Frieder Nake tried to ascertain the algorithmic sign is probably an argument in favor of Sieckenius de Souza's view. But Nake and Andersen refer to quasi-signs and see the interface as the *sender* (in Shannon's sense). Sieckenius de Souza made a major choice: Peirce! The reason is clear: recognition of the interpretant process.

On this note, a significant semiotic issue has to be acknowledged. Language partakes in human interaction in a variety of ways. But which language actually results in a human being able to cope with change? The language of scientists? The language of literature? The cultivated language disseminated through education? The reduced language (small vocabulary, rudimentary grammar, use of stereotypes, etc.) of an increasing number of the population? In terms of Sieckenius de Souza's focus on HCI—which interfaces help most: the dumbing down or the challenging?—important research (van Nimwegen & van Oostendorp, 2009) suggests interesting results that can inform the engineering aspects of the semiotics of HCI. Indeed, the more the interface replaces human effort, the lower the adaptivity of the user to new situations. This is, of course, also relevant to the semiotic study of programming.

Qui prodest?

The hope that cross-pollination is beneficial animates everyone who takes inter- and cross-disciplinarity seriously. Does computer science benefit from semiotics? Does semiotics benefit from computer science? Moreover, do we get a better understanding of ourselves facilitated by semiotic experiences of a nature different from those human beings had in the past? The semiotics associated with hunt and gathering, or that associated with agriculture, or with the industrial age is fundamentally different from that of the information age. (*Fundamental* means that there is a discontinuity that must be acknowledged.) Over time, many authors have addressed such questions. In reviewing Tanaka-Ishii's book, Kevin McGee (National University of Singapore) brought up the issue of "how semiotics and formal analysis inform each other" (McGee, 2011, p. 930). He is right in pointing out that authors of contributions to the semiotic aspects of communication "tend to be primarily either semioticians analyzing technology or technologists using semiotic concepts … to discuss technology"(McGee, p. 931). In this day and age of specialization, the hope that somebody could acquire competence in both knowledge domains is at best naïve. Tanaka-Ishii's training as a linguist makes her a good candidate for understanding formal languages. In addition, she does not pursue the subject in its very broad sense, but defines reflexivity as her focus, acknowledging that semiotics can effectively handle the subject. ("The Aim of This Book," subsection 1.1, is a very assuring declaration of integrity.) Without grand proclamations, she lays down very clear premises:

> " machine-based and human systems can be considered similar to some extent" (Tanaka-Ishii, 2010, p. 2).
>
> "application of semiotic theories to programming enables the consideration … of the universal and specific nature of signs in machine and human systems" (Tanaka-Ishii, 2010, p. 3; see her diagram on p. 3).
> "the difference between computer signs and human signs lies in their different capability to handle reflexivity." (Tanaka-Ishii, 2010, p. 3)

There are soft spots: In her discussion of similarity between human and machine, what is meant by "to some extent" and "What are signs in machines?" and so forth. But this is an honest book, focused to the extent that the author ignores anything that does not specifically serve her journey. She actually believes—rightly or wrongly—that almost nothing has been contributed to the domain before she started out. I prefer this parsimonious attitude to that of authors who make Wikipedia and Google their research sources. But the credit we owe this author for keeping things simple (including the sparse references) comes at a price: In her view, reflexivity within functional programming is related to Saussure's dyadic model, while Peirce's triadic structure corresponds to object-oriented programming. To ascertain is, of course, easier than to demonstrate or justify. She distinguishes between models of signs (corresponding to what signs are, see Tanaka-Ishii, 2010, p. 6), kinds of signs, and systems of signs. Aware of the fact that competence is a necessary premise, she tries to define it. And this is where things get a bit slippery. "When I began writing this book," we are informed, "semiotic theory was not sufficiently established to be straightforwardly applied in a complete form that could be introduced at the beginning of the book," (Tanaka-Ishii, p. 7).

This cannot be left unchallenged. Joseph Goguen, whose impressive work still awaits recognition, had already established his group at the University of California-San Diego.[2] Neither can the attempt to introduce hypotheses via works of art ("intuitive or metaphorical introduction," Tanaka-Ishii, 2010, p. 8) be taken seriously. (Again, Goguen was ahead, though working more with examples from music, so much better suited to discussions on programming.) A more comprehensive semiotic foundation would have informed Tanaka-Ishii that reproductions (especially in black and white) are not really what the original might mean; thus references to shades of grey are as telling of grey and art reproductions as any image (painting, photograph, drawing). The author is well educated, well read, interested in art, but less than precise

2. I had the opportunity to talk to him (at Stanford, we shared an interest in category theory) about his *Algebraic Semiotics* (1999), and we continued our dialog on programming until his untimely death.

in the analysis suggested. Semiotics and semiology are compared without a deep understanding of their respective conditions. Saussure is the master of the synchronic view; Peirce advances the dynamic view. One cannot write, "Saussure's signified corresponds to Peirce's object," (Tanaka-Ishii, p. 29) without risking simplifications that neuter the semiotic animal. Peirce's immediate object vs. the dynamic object is another distinction that should not be ignored. Moreover, the interpretant requires a totally different take than the one apparent in this book. Peirce's semiotic is part and parcel of his philosophy (of which, to her credit, Tanaka-Ishii is aware; see her discussion of Firstness, Secondness, Thirdness, pp. 104, 123). To be clear: Peirce's is a triadic-trichotomic semiotic conception. Within this understanding, one cannot write about applying the trichotomy to the computer signs (Tanaka-Ishii, p. 105), as one cannot take the types of representation (iconic, indexical, symbolic) and turn them into "kinds of signs." The kinds, in Peirce's view, are the ten classes he defined. Of course, operating with these ten classes would be cumbersome, but this is the only way to make sense of Peirce's comprehensive system. Reductions can become dangerous simplifications. The alternative is to contribute your own concepts.

Assuming that Tanaka-Ishii's book lays claim to being the first to attempt some semiotic foundation for programming, we realize that this is a high-order challenge, especially in view of the fact that while natural language is expressive, but not precise (see Nadin, 1997, pp. 161, 255-256, 264-269, 682), programming languages are expected to be precise to the extent of eliminating ambiguity (which machines cannot handle). But Tanaka-Ishii does not even address this defining aspect.

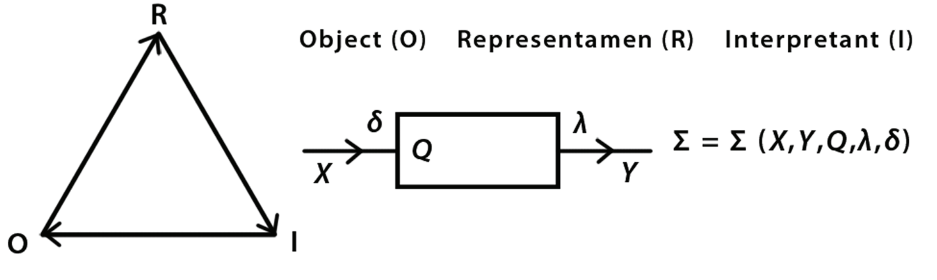
It would be wrong if readers interpret such observations as indicating that this book is not worth their effort. All I am trying to suggest is that this good book could have been better—exactly what we need if we want to make the point that semiotics is relevant. Tanaka-Ishii's competence in formal languages, programming languages, in particular, qualifies her as a promising researcher of semiotic implications of the age of computation. The self-referential character of human language, in which representation and interpretation are intertwined, has served humankind quit well. Computer-based activities rely on languages that are poor in terms of reflexivity. They were conceived for a different reason. This is a fundamental aspect: Can we extend the dynamics of the living, in particular its evolutionary characteristics, into the realm of machines? Can languages be the agent in achieving this result? The similarity between human and machine, which is one of her premises, has enticed many speculations. The fact that happiness (or emotions)—a state in the system we call *living*—lives in the space of ambiguity, and is not necessarily a machine-achievable goal might surprise some. But not Tanaka-Ishii, who is aware of reflexivity, but who is also a very opportunistic researcher. The book, a serious attempt at focusing on a small aspect of the problem, does not venture an answer. But if traces—remember, we looked into the history of semiotics for traces of questions posed at various times—are our goal, here we have the trace of a major question. It was formulated seriously, and the author deserves to be acknowledged for this, regardless of the shortcomings mentioned.

Nevertheless, the question posed is, in the final analysis, an extension of Hilbert's *Entscheidungsproblem* (1928)—the search for an algorithm for solving Diophantine equations, generalized to all equations. As such, Gödel and Turing already let us know that there is no such algorithm. If Tanaka-Ishii wanted to challenge their proof, she did not succeed. Having examined mathematical insight during a lecture (March 22, 1995), Martin Davis might see Tanaka-Ishii's arguments as validating his views today. But this is no longer a semiotic subject, rather one of logic. That Peirce wanted his semiotics to be a *logic of vagueness* (see Nadin, 1980, 1983) only shows how difficult it is to distinguish between logic and semiotics (not to say logic and mathematics).

The Semiotic Engine: A Discussion That Does Not Go Away

The silly thing about claims of "I was the first" is that respectable members of academia fail to realize that they describe the outcome of a competition in which everyone runs in a different direction. There is no real need to re-open the debate. Over the years, I argued with Kenneth Ketner (in 1988), Gerd Döben-Henisch (in 1995), Barbara da Silveira and, indirectly Winfried Nöth (2002) on matters of semiotics and computation. I also had substantive exchanges on the matter with the late Peter Bøgh Andersen and Frieder Nake (in 1992, 1994, and ongoing), as well as with Solomon Marcus (as recently as the summer of 2009). Initially, the notion of the semiotic engine (as I called it, inspired by the analytic engine terminology) came up in my discussions with Max Bense—a stubborn determinist, but also a congenial partner in dialog. In *Semiosis* (Nadin, 1977), I published the mathematical proof of the equivalence between the Peircean sign definition and fuzzy automata.

Figure 2: Representation and Fuzzy Automata: Proving the Equivalence of Formal Definitions.



Note: The notion of automaton is a generalization of machine functioning. The fuzzy description of input and output values corresponds to the intention of capturing not only quantities, but also qualities. The two transfer functions δ and λ could be defined in such a manner that pseudo-non-determinism behaviors can be simulated by the fuzzy automaton.

This inspired the analogy to an engine that embodies semiotic processes. Over time, I lost sight of my initial premise. It took until January 2010[3] to realize that various authors arguing in favor of the semiotic machine terminology are advancing an understanding of semiotics that is not mine. Indeed, only for those who equate information processes and semiotic processes does this expression appear as justified. Moreover, only for those who see no difference between the living and the physical does the reduction to the machine (going back to Descartes) make sense. This is not my case. The mathematical proof of the equivalence between the dynamic definition of the sign and fuzzy automata was, at best, a starting point for further elaborations that led me to anticipatory systems (anticipation).

The information processing machine embodies our understanding of the world of action-reaction. It does not know what uncertainty is, although it knows a lot about probability. The semiotic machine (more a formal analogy than a physical realization) is the expression of questions concerning anticipation. Such a "machine" is non-deterministic, has at least two clocks, and is highly adaptive and goal driven (teleological). With this in mind, I realize that the varieties of understandings associated with those I mentioned above corresponds to fundamental positions that are not reconcilable. In a remarkable contribution (Hong, 2007), Sungook Hong framed the subject of the relation between "Man and Machine in the 1960s" a very broad perspective. There is no need to quote here, at length, Norbert Wiener, Heinz von Foerster, Mansfield Clynes, Nathan Kline (to whom we owe the term *cyborg*), Erich Fromm, Jacques Ellul, Lewis Mumford, John Kenneth Galbraith, and others. But there is a definite need to become aware of the distinction that François Jacob (1974) made between the *physical world* and that of the *living*: one of information processes, the other with the added dimension of semiotic processes—and their inherent uncertainty. *Simple machines* (Rosen's terminology, 1985, p. 111) "do not make errors," complex machines can behave erroneously (disease is such a behavior). In its von Neumann-Turing embodiment, the computer is a deterministic machine. Different modes of computation are extremely seductive, but not yet available. The deterministic machine works below the threshold of complexity associated with the living (and with anticipation as a necessary characteristic of existence. Semiotic activity corresponds to complexity. Below the threshold of the living, we deal with pseudo-signs and not with signs (if you still want to focus on signs and not on representations).

Researchers of molecular computing (Tadashi Nakano of UC-Irving, Miles Pierce of Caltech, John Reif of Duke University, among others) entertain the hope of synthesizing *living machines* (see Kroeker, 2008). Turing's autonomous self-structuring seems appropriate to dealing with *living computation*. Embedded or not in human bodies, it is an expression of connecting metabolism and representation, that is, information and semiotic processes embodied in meaningful information.

Of course, before any other step ahead is made, we need to further define our terminology. In this sense, for me semiotics does not have as its subject the often

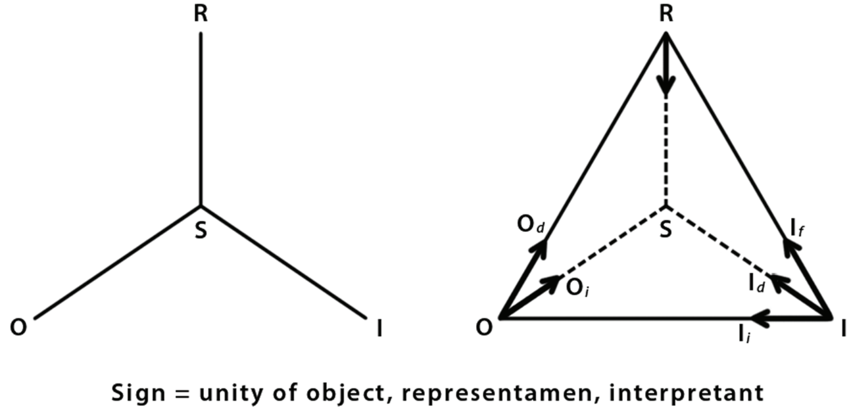
3. When I enjoyed a series of meetings at Frieder Nake's compArt group at the University of Bremen.

misused concept (entity) called *the sign*. Rather, focusing on the sign, defined in a variety of ways (some better justified than others), semiotics ends up as an ill-defined discipline. The characteristic of semiotics, as Hausdorff understood and as Cassirer argued for is re-presentation. The fact that the means of representation can be called *signs*, or be defined as signs, is less relevant than the essential functions of semiotics. In close relation to representation is the function of interpretation through which meaning is conjured. Moreover, representations are subject to further representation, to operations on representations, to interpretations that in turn become new representations, and so on. Sign is as good a placeholder as numbers are in mathematics. But no more than that. They do not relate to interactions, which are the main characteristic of the living. In particular, mind interactions are defining: We automatically read minds all the time, we probe the future, we are in anticipation (which is different from we *anticipate*, a verb that actually makes no sense).

One more detail: Semiotic reflexivity translates as semiotic awareness. Without awareness of the role that representations have in our understanding of the world and of ourselves, there is no semiotics. No machine ever came up with, on its own doing, with its own resources, a symbol (as Lewis Mumford pointed out: "No computer can make a new symbol out of its own resources" (1967, p. 29). Even if it could, it would not know what to do with it, how it can be interpreted.

The semiotic system that we associate with Peirce, in particular the interpretant process, comes closest to what I suggest. The Peircean definition of the sign, that is, the unity between the object (immediate and dynamic), the representation, and the interpretant, is actually the description of an infinite process, which is the semiosis.

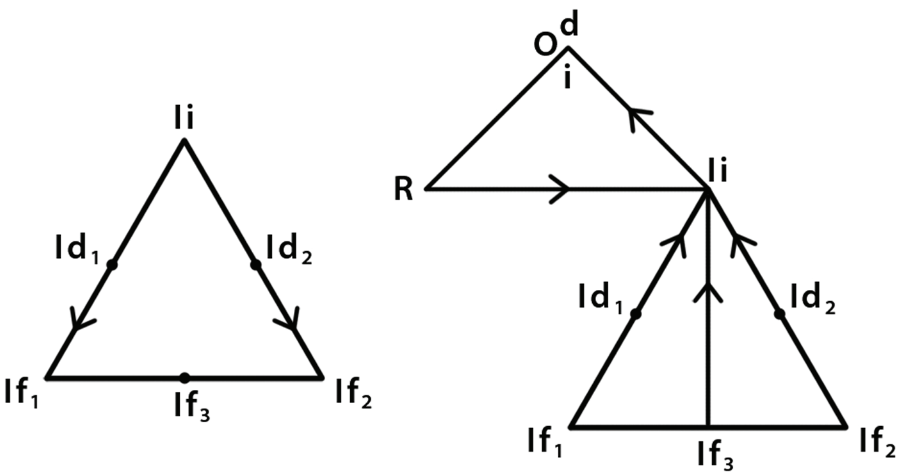
Figure 3: The Dynamics of Semiotic Processes Is Implicit in Peirce's Semiotic Conception



Note: The diagrams correspond to an understanding of representation as the unity between what is represented, the medium (as Peirce called the sign) of representation, and the process of interpretation. The object of representation is further differentiated as immediate and dynamic; the process of interpretation results in immediate, dynamic, and final interpretations (corresponding to theories).

Take note that once we acknowledge the object levels and the interpretant levels, the sign ceases to be a synchronic entity. It gets life, in the sense that the interpretant process injects dynamics into its reality. We never deal with signs; we always deal with representations, sign aggregates whose dynamic meaning is a function of the context, not of the alphabet (i.e., sign repertory).

Figure 4: Object and Interpretant Domain



Dynamic interpretation of the object and interpretant domain

Note: The diagrams make explicit the various levels of understanding of the object represented and of the process of interpretation.

But this is not a discussion about terminology, rather about the relevance of the discipline. The semiotic engine to which I referred processes representations and instantiates semioses. The consequences of this statement can be retraced in the books I discussed, as well as in the broader consideration of today's semiotic scene. Taken as printed books, each of the three titles discussed has its justification, if not necessarily an easily identifiable raison d'être. The scholarship is deplorable in some cases, in others, still timid, but promising. As testimony to what semioticians do, they are not encouraging. Of course, there is no obligation to be faithful to one or another author. This is not religion—dogmatism is a poor compass (no matter which direction it points to). But there is an obligation to preserve scientific integrity. Nobody will take seriously someone who changes Pythagoras' Theorem to fit some dubious purpose. If scientific arguments speak in favor of contradicting a theorem, so be it. Submit your own, with the appropriate arguments. But first you need to know it and understand it.

The Alphabet and Grammar of Computation

Deep down, in the digital engine, there are two elements controlling and making computation possible: an "alphabet" and a "grammar." These two together make up a language, that is, machine language. The alphabet consists of two letters (0 and 1). The grammar is the Boolean logic (slightly modified since Boole, but in essence a body of rules that make sense in the binary language of Yes and No in which our programs are written). The assembler—with a minimum of "words" and rules for making meaningful "statements"—comes on top of this machine language; and after that, the level of formal language performance, in which programs are written or automatically generated. Such programs need to be evaluated, interpreted, and executed. Here I submit to the reader structural details we all know (some in more detail than others) but which only rarely preoccupy us. My purpose is very simple: to lend meaning to my point that, in order to be meaningful, computers ought to be semiotic machines (an idea I first articulated over 30 years ago). Too many scholars took over my formulation (with or without quotation marks or attribution) without understanding that as a statement, it is almost trivial. What my colleagues—some of them respectable authors and active in semiotic and computer science organizations that assert their legitimacy—have totally missed is the need to realize that such a description makes sense only if it advances our understanding of what we describe. To say that the computer is a semiotic machine means to realize that what counts in the functioning of such machines are not electrons (and in the future, light or quanta or organic matter), but information and meaning expressed in semiotic forms, in programs, in particular, or, more recently, in apps. We take representations (which reflect the relation between what a sign represents and the way something is represented) and process them. Moreover, as we use computation, we try, after processing, to assign a meaning to our representation. Since in the machine itself, or in the program that is a machine, there is no place for a semantic dimension, we build ontologies (which are databases similar to encyclopedias or dictionaries) and effect association. This is how search engines frequently work; this is what stands behind the new verb *to Google* and our actions when we start a search by identifying sources of information in the World Wide Web.

The two-letter language (of zeros and ones) and the grammar (Boole's logic) allow for precision. Once we realize that we are not after information only, but after meaning as well, things get a bit more complicated. Actually, we do want to maintain precision, but we also seek expressiveness. The natural language alphabet (26 letters in the English Roman alphabet), along with grammar, made not only science, but also poetry, possible. Nobody in his (or her) right mind reads a poem in order to obtain information (expressed in bits and bytes), or for the sake of information. No one plays a computer game for the sake of information. Meaning is what the reader constructs in the interpretation or in the game action. The same holds true for interpreting the living computation—the meaning of change from a condition defined as healthy to a condition defined as diseased. Medicine focused exclusively on information fails exactly because it ignores the meaning of changes in the information. A computational medical diagnostic has to integrate both information and meaning.

Many Valued Logic as an Alternative

The extreme precision brought about by an alphabet of two letters and a grammar of clear-cut logic comes, as we have seen, at the expense of expressiveness. The more precise we are, the less expressive the result. Fuzzy set-based descriptions are much richer in detail; three-valued logic is by many orders of magnitude more productive than the two-valued Boolean logic. Fuzzy logic is even more supportive of rich expression. In terms of program timeliness—future-ness, in particular—this means that we could capture time and make it a part of programs only, and only, when computation will transcend, as it partially does, not just its syntactic dimension, but also the semantic dimension of the signs making up programming languages. Indeed, at the moment when computation will be pragmatically driven, that is, by what we do, it will acquire a time dimension coherent with our own time (Nadin, 2011). And it will reflect the variability of time. Interestingly enough, this is partially happening in the computation required by interactive massively multiplayer online role playing games (MMORPG).

The most recent Workshop on "Semiotics, Cognitive Science, and Mathematics" (at the renowned Fields Institute, March 11-14, 2011) prompted one of the speakers to bring up an image of semiotics that he could easily refute: Semiotics is considered as an anachronistic academic field, the same as Philology and Egyptology (Neuman, 2011). I am not surprised. The lack of competence—I refer to those real or imaginary characters alluded to—is what undermines semiotics. Fish start rotting at the head.

Instead of a Conclusion

As I was finishing this article, a new periodical made its debut: *The International Journal of Signs and Semiotic Systems*. This is a significant development: A new generation takes over, our students advance a research agenda that breaks with the past. We read about the emergence and evolution of semiotic processes, information interpretation systems, embodied and situated semiotic processes, sign and symbol grounding, biologically inspired models of semiotic systems, and much more. This is a high-order agenda, and the two Editors-in-Chief—Angelo Lula and João Queiroz—are dedicated to a forum within which discussions should take place. For reasons of disclosure, I need to add that I was invited to be on the editorial board, and I hope to contribute to broadening the agenda. Of course, the understanding of the complementary nature of information and meaning is of extreme relevance.

An entire book (Brier, 2008), dedicated to cybersemiotics, makes the point: *Why Information is not Enough!* In more words than I felt entitled to use in a review article, the point is made that understanding the living is a prerequisite for formulating a coherent theory of semiotics. I could, of course, make use of some of Brier's formulations, but for some reason, I prefer to exemplify the thesis of this article (and my view on the relation between information theory and semiotics) quoting Einstein: "It would be possible to describe everything scientifically, but it would make no sense; it would be without meaning, as if you described a Beethoven symphony as a variation of wave pressure." With Einstein's tacit approval, I would add: The same applies to describing Beethoven's symphony in zeros and ones.

Acknowledgments

The author expresses gratitude to Winfried Nöth for getting me started on writing this review article. Søren Brier contributed questions I wish I came up with. Jeanette Bopry made the editorial process as meaningful as it can be. If errors persisted, I take full responsibility.

References

Andersen, P. B. (1990). *A theory of computer semiotics*. New York: Cambridge University Press.

Andersen, P. B. (2003). The digital age: Cohesion and coherence in programs. In M. Vilanova & F. Chorda (Eds.), *A mind at work. Festschrift Mihai Nadin* (pp. 7-20). Heidelberg: Synchron Publishers. (The text is available at http://www.nadin.ws/archives/416)

Aristotle (1961). *Poetics*. (S. H. Butcher, Trans.). New York: Hill and Wang. Retrieved March 31, 2011 from, http://classics.mit.edu/Aristotle/poetics.html (Aristotle's original: 350 BCE)

Arnaud, A. Nicole, P., & Fouillée, A. (1964). *The art of thinking; Port-Royal logic*. New York: Bobbs-Merrill. (The 1878 edition, *La logique de Port Royal*, is available online: Retrieved June 7, 2011 from http://www.archive.org/details/lalogiquedeport00fouigoog

Augustine. (1958). *De doctrina cristiana* (D.W. Robertson, Trans). Upper Saddle River, NJ: Prentice Hall. (Augustine's original: 397 AD)

Bateson, Gregory (1972). *Steps to an ecology of mind: Collected essays in anthropology, psychiatry, evolution, and epistemology.* Chicago: University Of Chicago Press.

Brier, S. (2008). *Cybersemiotics: Why information is not enough!* Toronto: University of Toronto Press.

Conery, J. S. (2010). What is computation? Computation is symbol manipulation. *Ubiquity*, November. Retrieved on April 27, 2011 from, http://ubiquity.acm.org/article.cfm?id=1889839 (*Ubitquity* symposium "What is Computation?")

Davies, C. W. P. (2004). John Archibald Wheeler and the clash of ideas. In J. D. Barrow, C. W. P. Davies, & C. L. Harper (Eds.), *Science and ultimate reality* (pp. 3-24). London: Cambridge University Press.

Einstein, A. (n.d.). *In a conversation with Segré.* Retrieved April 21, 2011 from http://thinkexist.com/quotation/it_would_be_possible_to_describe_everything/15520.html

Gallese, V. (2001). The "shared manifold" hypothesis. From mirror neurons to empathy. *Journal of Consciousness Studies*, *8* (5-7), 33-50.

Goguen, J. (1999). An introduction to algebraic semiotics, with application to user interface design. In C. Nehaniv (Vol. Ed.), *Lecture notes in computer science: Vol. 1592. Computation for Metaphors, Analogy, and Agents* (pp. 242-291). Heidelberg: Springer.

Hilbert, D., & Ackermann, W. (1928). *Grundzüge der theoretischen Logik* (Principles of mathematical logic). Berlin: Springer-Verlag,

Hobbes, T. (2010). *Leviathan: Or the matter, forme, and power of a common-wealth ecclesiasticall and civill*. Shapiro, I. (Ed.). New Haven, CT: Yale University Press. (Leviathan was originally published in 1650)

Hong, S. (2007). Man and machine in the 1960s. *Techné: Research in Philosophy and Technology, 11* (1). Retrieved June 6, 2011 from http://scholar.lib.vt.edu/ejournals/SPT/v7n3/hong.html

Jacob, F. (1974). *Logic of living systems: History of heredity* (B. Spillman, Trans.). London: Allen Lane.

Kroeker, K. L. (2008). Living machines. *Communications of the Association for Computing Machinery (CACM), 15* (12), 11-13. (Also available online [retrieved June 5, 2011] at http://kroeker.net/published/living-machines.htm)

Lambert, J. H. (1764). Neues Organon oder Gedanken über die Erforschung und Bezeichnungdes Wahren und dessen Unterscheidung vom Irrtum und Schein. Leipzig: Johann Wendler.

Liu, K. (2005). *Semiotics in information systems engineering*. Cambridge: Cambridge University Press. (This reprint of *Semiotics in information systems engineering,* 2000, Cambridge: Cambridge University Press was distributed in China.)

Livio, M. (2003). *The golden ratio: The story of PHI, the world's most astonishing number.* New York: Random House.

Locke, J. (1979). Of the division of the sciences. In P. H. Nidditch (Ed.), *An Essay Concerning Human Understanding* (chapt. XXI). New York: Oxford University Press. (Originally published in 1690) Also available online http://enlightenment.supersaturated.com/johnlocke/BOOKIVChapterXXI.html [retrieved June 5, 2010].

McGee, K. (2011). K. Tanaka-Ishii: Semiotics of programming (Book review). *Artificial Intelligence*, *175* (5-6), 930-931.

Mitchell, P. J. (2009). Watching minds interact. In M. Brockman (Ed.), *What's next: Dispatches on the future of science* (pp. 78-88*)*. New York: Vintage.

Mongré, P. (1897). *Sant' Ilario, Thoughts from Zarathustra's Country* (*St. Ilario – Gedanken aus der Landschaft Zarathustras*). Leipzig: C. G. Nauman. (This was the first book that Felix Hausdorff published under the pseudonym of Paul Mongré.)

Mumford, L. (1967). *The myth of the machine: Technics and human development*. New York: Harcourt, Brace, Jovanovich.

Nadin, M. (1977). Sign and fuzzy automata. *Semiosis, 1* (5). Text available online as pdf: http://www.nadin.ws/archives/395

Nadin, M. (1980). The logic of vagueness and the category of synechism. *The Monist, 63* (3), 351-363. (Also available in E. Freeman (Ed.), *The relevance of Charles Peirce* (pp. 154-166). La Salle, IL: The Monist Library of Philosophy, 1983.)

Nadin, M. (1982). Consistency, completeness, and the meaning of sign theories. *The American Journal of Semiotics*, *1* (3), 79-88.

Nadin, M. (1991). *Mind—Anticipation and chaos*. (In the series: Milestones in Research and Discovery). Stuttgart: Belser. (See also http://www.oikos.org/naminds1.htm)

Nadin, M. (1997). *The civilization of illiteracy*. Dresden: Dresden University Press.

Nadin, M. (2011). Computation, information, meaning. Anticipation and games. *International Journal of Applied Research on Information Technology and Computing*, *2* (1), 1-33.

Neumann, J. von (1963). The general and logical theory of automata. In A. H. Taub (Ed.), *John von Neumann Collected Works, Vol. 5* (pp.288-326). New York: Pergamon. (Originally presented at the 1948 Hixson Symposium, California Institute of Technology, September 20, 1948)

Neuman, Y. (2011). *Semiotics, mathematics & information technology: The future is already here*. Retrieved on April 4, 2011 from http://www.fields.utoronto.ca/programs/scientific/10-11/semiotics/abstracts.html#marcus

Nimwegen, C. van, & Oostendorp, H. van (2009). The questionable impact of an assisting interface on performance in transfer situations. *International Journal of Industrial Ergonomics, 39* (3), 501-508.

Nöth, W. (2002). Semiotic machines. *Cybernetics & Human Knowing*, 9 (1), 5-21.

Paul, H. (1880). *Prinzipien der Sprachgeschichte*. Halle: Max Niemeyer. (Published in English as *Principles of the history of language* [H.A. Strong, Trans.]. College Park: McGroth Publishing Company, 1970.)

Peirce, C. S. (1887). Logical machines. *The American Journal of Psychology, 1* (1), 164-169.

Plato (2010). *Cratylus* (B. Jowett, Trans.). Rockville, MD: Wildside Press. Retrieved on April 10, 2011 from, http://classics.mit.edu/Plato/cratylus.html

Rosen, R. (1978). *Fundamentals of measurement and representation of natural systems*. Amsterdam: Elsevier Science.

Rosen, R. (1985). Organisms as causality systems which are not machines: An essay on the nature of complexity. In R. Rosen (Ed.), *Rosen: Theoretical biology and complexity* (pp. 165-203). Orlando, FL: Academic Press.

Saussure, F. de (1968). *Cours de linguistique générale*, *Vol. I*. Wiesbaden: Otto Harrassowitz.

Shannon, C. E. (1953). Computers and automata. *Proceedings of the Institute of Radio Engineers, Oct. 1953, 41* (10), 1234-1241.

Shannon, C. E. (1993). Programming a computer for playing chess. In N. J. A. Sloane, & A. D. Wyner (Eds.), *Claude Elwood Shannon. Collected papers* (pp.657-666). Piscataway, NJ: IEEE Press. (First presented at the National IRE Convention, March 9, 1949.)

Shannon, C. E. & Weaver, W. (1949). *The mathematical theory of communication*. Urbana, IL: University of Illinois Press.

Sieckenius de Souza, C. (2005). *The semiotic engineering of human-computer interaction.* Cambridge, MA: The MIT Press.

Sieckenius da Souza, C. & Leitão, C. (2009). *Semiotic methods for scientific research in HCI*. Princeton, NJ: Morgan & Claypool.

Silveira, L. F. B. da (1993). Charles Sanders Pierce e a contemporânea filosofia da ciência: uma difícil conversação. *Revista Trans/Form/Ação*, *16*, 63-82. (See also *Some considerations about semiotic machines from the point of view of Charles S. Peirce's Philosophy.* Retrieved on April 15, 2011 from, http://www.inm.de/kip/semiotic/silveira_article.html)

Stamper, R. K. (1973). *Information in business and administrative systems*. London: B. T. Batsford.

Tanaka-Ishii, K. (2010). *The semiotics of programming.* New York: Cambridge University Press.

Williams, T. (2007). *Anselm: Basic writings.* Indianapolis: Hackett Publishing Company.

Zemanek, H. (1966). Semiotics and programming languages. *Comunications of the ACM*, *9* (3), 139-143.