Memory

Warren Sack

The following examination of computer memory closely scrutinizes the words, rhetoric, and discourse of computer science and several associated disciplines. Presupposed by this methodology of rhetorical analysis is the idea that the words employed in the design and evaluation of new technologies shape the form and function of those technologies. Of course, designers' vocabularies do not completely determine what a technology can do or how it works. After all, designers are not magicians and the activity of software design is not a form of incantation! But, many technologies were written and spoken about long before they were developed into practical, everyday things: flying machines and long distance communication are two technologies that were dreamt about long before they were implemented. Here we review a short history of the metaphors and analogies employed by philosophers, scientists, and technologists to understand memory. We will see how previous metaphors are sometimes later taken for literal truth. When metaphors become scientific models, alternative ways of thinking about the object of study become difficult. The purpose of this entry is to question the metaphors of memory taken as models and, thereby, begin to explore new ways to think about computer memory.

> The act of perception stamps in, as it were, a sort of impression of the percept, just as persons do who make an impression with a seal. This explains why, in those who are strongly moved owing to passion, or time of life, no mnemonic impression is formed; just as no impression would be formed if the movement of the seal were to impinge on running water; while there are others in whom, owing to the receiving surface being frayed, as happens to the stucco on old chamber walls, or owing to the hardness of the receiving surface, the requisite impression is not implanted at all.[1]

Aristotle's image of memory is constructed from a seal that is known to work on soft wax or clay. His presupposition is that when our memories are in working order they are akin to a pliant solid, like wax, that can record the impression of a seal.

Aristotle's trope does not begin or end with him. Plato wrote of the analogy before Aristotle; and, Cicero, Quinitilian, Sigmund Freud, and Jacques Derrida explored the trope of memory-as-wax-tablet after him. Each new generation of memory theorists tends to incorporate the latest media technology to explore its similarities with human memory. Or, to phrase this point polemically, as media theorist Friedrich Kittler and his followers have done for the past couple of decades, "Media, then, are [at] the end of theory because in practice they were already there to begin with."[2]

Historically, theorists have not always been clear about when their references to media technology are metaphorical and when they are literal. Derrida, for example, closely scrutinizes Freud's mixed and unstated metaphors about memory.[3] But, many of today's memory theorists quite clearly state that what others might take to be a metaphor, they take to be a literal truth. Contemporary theorists compare human memory and computer memory. Cognitive scientists who explore this analogy believe that humans and machines are two species of the same genus; in the words of computer scientist and economist Herbert Simon, humans and computers are "symbol systems."[4] Thus, cognitive scientists hypothesize that human memory is not akin to computer memory, it is virtually the same thing as computer memory. Or, to put it a different way, the hypothesis is that computer memory is not just one possible model of human memory, it is the best model of memory.

This belief, that the computer is the best model of the object of study, is not unique to cognitive science. It is an operating principle in molecular biology, operations research, neuro-psychology, immunology, game theory, economics, and many other sciences. Historian of science Philip Mirowski calls this literal belief in computation one of the defining characteristics of a "cyborg science," a science that does not use the computer as an analogy but which uses it as a simulacrum of the object of study.[5] For example, Howard Gardner, in his overview and introduction to cognitive science, states that one of the paramount features of cognitive science is this belief:

> There is the faith that central to any understanding of the human mind is the electronic computer. Not only are computers indispensable for carrying out studies of various sorts, but, more crucially, the computer also serves as the most viable model of how the human mind functions.[6]

The first set of models devised by cognitive psychologists to explain the structure and dynamics of human memory recapitulated many architectural aspects of then-contemporary computational hardware. For example, the model of Richard Atkinson and Richard Shiffrin[7] included a "short-term store," a "long-term store," "buffers," "slots," and a hypothesis that information processing for storing and retrieving items from memory was a sequential (rather than a parallel) operation. These are architectural details that one can also identify with the computers of that time (i.e., the 1960s). As work in this area developed, the memory models began to look less and less like then-contemporary computer hardware, but they are still frequently phrased in terms that would allow one to implement them in software.

What makes this tight coupling between human memory and computer memory seem plausible? Why might computer memory be seen as "the most viable" model of human memory? To untangle this belief of cognitive scientists it is necessary to remember that before computers were machines they were people, usually women. For over two hundred years, these women—these computers—worked together in groups compiling tables of statistics, tables of trigonometric functions, tables of logarithms. For example, computers worked together in 1757 to calculate the return trajectory of Halley's comet.[8]

When the machines we now call computers were first designed, they were designed to do the work of a human computer. In 1936, Alan Turing designed a machine that could do the work of a human computer. In his paper he writes of "computers" but when he does he is referring to those people who held the job of computer. Turing himself did not go so far as to say that his machine has memory, but he almost does. His mathematical paper is based on an extended analogy between a machine and a person, that is, a human computer. Turing explains how his machine might remember what it is doing and what it is to do next by extending the analogy like this:

> It is always possible for the computer to break off from his work, to go away and forget all about it, and later to come back and go on with it. If he does this he must leave a note of instructions . . . explaining how the work is to be continued. . . . We will suppose that the computer works by such a desultory manner that he never does more than one step at a sitting. The note of instructions must enable him to carry out one step and write the next note. Thus the state of progress of the computation at any stage is completely determined by the note of instructions and the symbols on the tape.[9]

Part of Turing's accomplishment was to show that these so-called "notes," the mnemonics for remembering what to do next, could, in general, always consist of a series of integers written on a paper tape. So from Aristotle's seals we have moved to a newer technology of bureaucracy, namely numbered paper forms.

During the World War II Turing's mathematical, theoretical machines became practical. The first computers had to be "set up" for each new problem of calculation. "Set up" entailed plugging and unplugging cables and setting hardware switches. By the end of the war, it became clear to J. Presper Eckert, John Mauchly, and John von Neumann that the memory of the computer could be used to store a program as well as data and that the program could be specified to automatically set up the computer to solve a new problem. Once the so-called "stored-program" memory was implemented computers could be programmed rather than "set up."[10]

These first computers were implemented in vacuum tubes and electronics and, from then on, the term "computer" meant a machine, not a human being. Ten years after Turing's publication there existed machines that were called "computers" and these computers were said to have memories.[11] Since many of the designers and builders of these first computers were engineers; and, since engineers had been writing, at least since the end of the a nineteenth century of the "magnetic memory" of iron;[12] and, since the physical substrate of early computer's "memories" was ferromagnetic,[13] this usage of the term "memory" to refer to the storage capacity of the computer is perhaps not so surprising. What is surprising is what happened next in the scientific world. Remember that social science, especially psychology, in the United States was dominated by behaviorism for most of the first half of the twentieth century. As Sherry Turkle puts it,

> As recently as the 1950s behaviorism dominated American academic psychology, its spirit captured by saying that it was permissible to study remembering but considered a violation of scientific rigor to talk about "the memory." One could study behavior but not inner states.

Turkle argues that

> The computer's role in the demise of behaviorism was not technical. It was the very existence of the computer that provided legitimation for a radically different way of seeing mind. Computer scientists had, of necessity, developed a vocabulary for talking about what was happening inside their machines, the "internal states" of general systems. If the new machine "minds" had inner states, surely people had them too. The psychologist George Miller, who was at Harvard during the heyday of behaviorism, has described how psychologists began to feel embarrassed about not being allowed to

> discuss memory now that computers had one . . . The computer presence relegitimated the study of memory and inner states within scientific psychology.[14]

What Turkle leaves out of her short history is that in 1956, when George Miller and his colleagues were founding the discipline of cognitive psychology, it had only been a few years since computers were not machines, but people. In other words, contemporary, cognitive science work on memory is based—ironically enough—on a willful amnesia of recent history and thus on a circularity: computer memory seems to be a good model of human memory because computer memory was modeled on human memory!

Here is the best analogy to the current situation that exists in many academic disciplines, many "cyborg sciences," that human thinking, memory, and decision making can be "modeled" by computer programs. This situation would be like discovering a painted portrait of a specific man and then spending the rest of one's professional life commenting on how uncanny it was that the portrait seemed to look like a human being.

The human that serves as the model for these cyborg sciences is culturally coded in a very specific manner. The human is, as Turing's analogy makes clear, not just any human. He—for, despite the fact than many human computers were women, it is usually a "he" in this technical literature—is a bookkeeper, accountant, or bureaucrat:

> We may compare a man in the process of computing a real number to a machine which is only capable of a finite number of conditions . . . The machine is supplied with a "tape" (the analogue of paper) running through it, and divided into sections (called "squares") each capable of bearing a "symbol." At any moment there is just one square . . . which is "in the machine." We may call this square the "scanned square." The symbol on the scanned square may be called the "scanned symbol." The "scanned symbol" is the only one of which the machine is, so to speak, "directly aware." However, by altering its m-configuration the machine can effectively remember some of the symbols which it has "seen" (scanned) previously.[15]

Here then is the true picture of the "human" that is the model for computer memory: he is a bureaucrat squirreling around in the back office, shuffling through stacks of gridded paper, reading, writing, and erasing numbers in little boxes. This Bartleby-the-Scrivener is the man so many cyborg scientists would like to portray or recreate as an assemblage of computational machinery.

Equipped with a clear picture of whose memory computer memory is designed to resemble, it becomes possible to parse the technical literature on computer memory. The technical literature is completely preoccupied with the management and allocation of memory. Memory in the technical literature is not Marcel Proust's lost aristocratic memories of, for instance, eating scallop-shell-shaped, lemon-and-butter-flavored cakes (madeleines) as a child. No, this technical literature is filled with the memories of bureaucrats: numbers, lists, tables, cells, and segments. Even the computer science literature on narrative memories boils down to a set of techniques for fitting stereotypical stories into preconceived grids.[16]

Memory, of this bureaucratic, gridded kind, is a major area of work in hardware and software research and development. It is easy to see the grid when examining hardware. For example, contemporary, dynamic random access memory (DRAM) consists of a matrix of capacitors—which either hold (1) or do not hold (0) a charge—wired together in rows and columns. At the lower levels of software (i.e., in the memory management routines of operating systems, programming languages, etc.) memory is represented as a vector (i.e., a fixed length sequence of integers) or a matrix (i.e., a vector of vectors) that can be indexed by row and column.

If one reads the canonical texts of undergraduate, computer science education one finds passages like this are ubiquitous to the writings about computer memory:

> Memory is an important resource that must be carefully managed. . . . The part of the operating system that manages memory is called [outrageous as it may seem!] the memory manager. Its job is to keep track of which parts of memory are in use and which parts are not in use, to allocate memory to processes when they need it and deallocate it when they are done, and to manage swapping between main memory and disk when main memory is not big enough to hold all of the processes.[17]

The function of a memory manager is akin to an accountant preparing taxes on his desk. If we understand his desk to be analogous to main memory and his file drawers to be like the computer's disk, then "memory allocation" is akin to assembling together the files and folders for a given account and finding space for them on the desk; "swapping" is like moving files and folders onto the desk from the file cabinets or, vice versa. The "resource" to be "managed" is the working space on the desk. Files and folders can be stacked, heaped, moved off the desk into file cabinets (i.e., onto disk), etc.

Undergraduate computer science students learn in their first or second year of studies the exact definitions and typical implementations of software analogs of "files," "folders," or "directories"; "stacks," "heaps," and "lists"; and the "recycling" or "garbage collection" of memory. Any adequate, introductory textbook on data structures and algorithms can provide the exact definitions of these "memory structures" and their associated operations.[18]

That these operations correlate almost exactly with what the bureaucrat does with his file cabinets, desk, and trash can is no coincidence. Neither is it a coincidence that these same operations are the ones available to today's computer users, whose graphical user interfaces are based on the so-called "desktop metaphor." The metaphors of the desk, the trash can, and the mind-numbing operations of office work and bureaucracy are built right into the foundations of the computer and its user interface. Even a quick skim through the seminal, foundational texts of graphical user interface design, especially those of Douglas Engelbart, make it clear that shuffling through, stacking, listing, and filing were the ideals of "memory" and "thought" admired and implemented by the founders of computer science and interface design.[19]

Of course, not all computing can be understood as office work. Rather, all computing is deeply rooted in the metaphors and pragmatics of bureaucracy; just as it is also intertwined with a genealogy of military thinking and materiel.[20] When these genealogies of software are forgotten, one loses sight of the highly particular and ultimately idiosyncratic images of memory and reasoning that are reified in the design and design principles of software.

Computer science's notion of "memory," that is, the "memory" of software and hardware, is not necessarily "worse" than that of other fields that investigate the issue of memory. But, computer science's working theories of memory are very specific and idiosyncractic to the concerns of bureaucracy, business and the military. This is largely because funding for computer science has come from these sources.

Juxtaposition with very different images of memory help one to imagine alternatives to the "closed world"[21] conditions that contemporary computational models circumscribe. For example, Marcel Proust's image of memory does not provide a better model of memory than the computer model, but it does provide a different model: a contrasting image that can be seen to highlight issues, ideas, and materialities uncommon to the military-(post)industrial technologies of memory:

And suddenly the memory revealed itself. The taste was that of the little piece of madeleine which on Sunday mornings at Combray . . . when I went to say good morning to her in her bedroom, my aunt Léonie used to give me, dipping it first in her own cup of tea or tisane. . . . when from a long-distant past nothing subsists, after the people are dead, after the things are broken and scattered, taste and smell alone, more fragile but more enduring, more unsubstantial, more persistent, more faithful, remain poised a long time, like souls, remembering, waiting, hoping, amid the ruins of all the rest; and bear unflinchingly, in the tiny and almost impalpable drop of their essence, the vast structure of recollection.[22]

Notes

1. Aristotle, *On Memory and Reminiscence*.

2. Geoffrey Winthrop-Young and Michael Wutz, "Translator's Introduction: Friedrich Kittler and Media Discourse Analysis," in Friedrich Kittler, *Gramophone, Film, Typewriter*, xx.

3. Jacques Derrida, "Freud and the Scene of Writing," in *Writing and Difference*, translated by Alan Bass.

4. Herbert A. Simon, *The Sciences of the Artificial*.

5. Philip Mirowski, *Machine Dreams: Economics Becomes a Cyborg Science*, 14.

6. Howard Gardner, *The Mind's New Science: A History of the Cognitive Revolution*, 6.

7. Richard Atkinson and Richard Shiffrin, "Human Memory: A Proposed System and Its Control Processes," in K. W. Spence and J. T. Spence, eds., *The Psychology of Learning and Motivation: Advances in Research and Theory, Volume 2*.

8. David Grier, *When Computers Were Human*, 19.

9. Alan Turing, "On Computable Numbers with an Application to the *Entscheidungsproblem*."

10. Paul E. Ceruzzi, *A History of Modern Computing*, 20–21.

11. Oxford English Dictionary, example from entry for "memory": "1945 J. P. ECKERT et al. Descr. ENIAC (PB 86242) (Moore School of Electr. Engin., Univ. Pennsylvania) iii. 1 The memory elements of the machine may be divided into two groups the 'internal memory' and the 'external memory.'"

12. Oxford English Dictionary, examples from entry for "memory": "1887 Jrnl. Soc. Telegr. Engin. 16 523 No matter how treated, a piece of soft iron has a 'magnetic memory.' 1935 Proc. Royal Soc. A. 149 72 The [magnetic] field..is to be regarded as 'frozen in' and represents a permanent memory of the field which existed when the metal was last cooled below the transition temperature."

13. The ENIAC used core memory: "Magnetic core memory, or ferrite-core memory, is an early form of computer memory. It uses small magnetic ceramic rings, the cores, to store information via the polarity of the magnetic field they contain. Such memory is often just called core memory, or, informally, core." Wikipedia: http://en.wikipedia.org/wiki/Core_memory; see also, Ceruzzi, *A History of Modern Computing,* 49–50.

14. Sherry Turkle, "Artificial Intelligence and Psychoanalysis: A New Alliance," *Daedalus* 17, 1 (Winter 1988).

15. Turing, "On Computable Numbers," 231.

16. Message Understanding Conference Proceedings (MUC-7), available at http://www.nlpir.nist.gov/related_projects/muc/proceedings/muc_7_toc.htm/ (last accessed April 9, 2006).

17. Andrew S. Tanenbaum, *Operating Systems: Design and Implementation*, 191.

18. For example, Alfred V. Aho, Jeffrey D. Ullman, and John E. Hopcroft, *Data Structures and Algorithms.*

19. Douglas Engelbart, "Augmenting Human Intellect: A Conceptual Framework," Summary Report for SRI Project No. 3578; see, especially, p. 56. Also, available online at http://www.bootstrap.org/augdocs/friedewald030402/augmentinghumanintellect/3examples.html#A.3 (consulted on April 9, 2006).

20. Manuel De Landa, *War in the Age of Intelligent Machines*.

21. Paul Edwards, *The Closed World: Computers and the Politics of Discourse in Cold War America*.

22. Marcel Proust, *Remembrance of Things Past. Volume 1: Swann's Way: Within a Budding Grove*, translated by C. K. Scott Moncrieff and Terence Kilmartin, 50.

■