Killer Apps and Technomyths

BY JANET ABBATE*

Paul Dourish and Genevieve Bell. *Divining a Digital Future: Mess and Mythology in Ubiquitous Computing.* Cambridge, MA: MIT Press, 2011. x + 248 pp., illus., index. ISBN 978-0-262-01555-4. $32.00 (hardcover).

Michael Sean Mahoney. *Histories of Computing.* Ed. Thomas Haigh. Cambridge, MA: Harvard University Press, 2011. x + 250 pp., illus., index. ISBN 978-0-674-05568-1. $49.95 (hardcover).

Eden Medina. *Cybernetic Revolutionaries: Technology and Politics in Allende's Chile.* Cambridge, MA: MIT Press, 2011. xvi + 326 pp., illus., index. ISBN 978-0-262-01649-4. $32.00 (hardcover).

Joseph A. November. *Biomedical Computing: Digitizing Life in the United States.* Baltimore, MD: Johns Hopkins University Press, 2012. xvi + 344 pp., illus., index. ISBN 978-1-421-40468-4. $60.00 (hardcover).

Sherry Turkle. *Simulation and Its Discontents.* Cambridge, MA: MIT Press, 2009. xiv + 217 pp., illus., index. ISBN 0-262-01270-7. $23.95 (hardcover).

Half a century ago, the popular icon of computing was a room-filling IBM machine. In the 1980s, computing became personal and Microsoft's rise reflected the growing importance of software. Today, the popular imagination links computing with social media and portable devices that offer an app for every need or whim. Throughout these changes in hardware, software, and users, a seeming constant has been the utopian social visions projected onto computer technology. Liberation, democracy, prosperity, community: all just a click away. Historians of computing have mirrored these shifts in their own

*Dept. of Science and Technology in Society, Virginia Tech Northern Virginia Center, 7054 Haycock Road, Falls Church, VA, 22046; abbate@vt.edu.

Historical Studies in the Natural Sciences, Vol. 43, Number 1, pps. 105–112. ISSN 1939-1811, electronic ISSN 1939-182X. DOI: 10.1525/hsns.2013.43.1.105.

work. Initially this subfield focused on groundbreaking machines and the individuals who created them. In the late 1990s historians somewhat belatedly turned serious attention to software, users, and application areas. At the same time, they have been increasingly engaged with the larger political and cultural contexts from which utopian visions of computing have sprung.[1]

Histories of Computing, a posthumous anthology of Michael S. Mahoney's essays spanning 1988–2008, conveys a sense of this historiographic trajectory. Readers unfamiliar with Mahoney should turn first to the Éloge to learn of his career as a historian of mathematics and the Scientific Revolution before he turned his attention to computing and became an influential voice in that emerging subfield. Thomas Haigh's opening essay pulls out important themes in Mahoney's work, such as connections across disciplines and the development of scientific communities, and notes the influence of Mahoney's advisor, Thomas Kuhn.

Mahoney's own essays are divided into three groups. The first set is historiographic: Mahoney exhorts historians of computing to ask deeper contextual questions and to pay more attention to software as artifact, industry, and occupation. "The Histories of Computing(s)" (2005) articulates a manifesto for "decentering the machine" and focusing instead on the communities of practitioners and users who bring their own histories and agendas to computing (57). The second set of essays, on the history of software engineering, highlights social contexts and power relations. "Software: The

1. Standard general works on the history of computing include Martin Campbell-Kelly and William Aspray, *Computer: A History of the Information Machine* (New York: Basic Books, 1996); Paul E. Ceruzzi, *A History of Modern Computing* (Cambridge, MA: MIT Press, 1998); Atsushi Akera and Rik Nebeker, eds., *From 0 to 1: An Authoritative History of Modern Computing* (New York: Oxford University Press, 2002). Volumes devoted to software include Ulf Hashagen, Reinhard Keil-Slawik, and Arthur Norberg, eds., *History of Computing: Software Issues* (New York: Springer, 2002); and Martin Campbell-Kelly, *From Airline Reservations to Sonic the Hedgehog: A History of the Software Industry* (Cambridge, MA: MIT Press, 2003). Works explicitly engaged with the politics and cultural meaning of computing include Paul N. Edwards, *The Closed World: Computers and the Politics of Discourse in Cold War America* (Cambridge, MA: MIT Press, 1996); Paul N. Edwards, *A Vast Machine: Computer Models, Climate Data, and the Politics of Global Warming* (Cambridge, MA: MIT Press, 2010); Rebecca Slayton, "Speaking as Scientists: Computer Professionals in the Star Wars Debate," *History and Technology* 19, no. 4 (2003): 335–64; Donald MacKenzie, *Mechanizing Proof: Computing, Risk, and Trust* (Cambridge, MA: MIT Press, 2001). Historians of computing were influenced by the "user turn" in technology studies exemplified by Ruth Schwartz Cowan, "The Consumption Junction," in *The Social Construction of Technological Systems*, ed. Wiebe E. Bijker, Thomas P. Hughes, and Trevor Pinch (Cambridge, MA: MIT Press, 1987), 261–80; and Nelly Oudshoorn and Trevor Pinch, eds., *How Users Matter: The Co-Construction of Users and Technology* (Cambridge, MA: MIT Press, 2003).

Self-Programming Machine" (2002) describes software engineering as a series of attempts to control workers. Mahoney notes how the factory discipline of Taylor and Ford "was translated into such concepts as 'mass-produced software components,' modular programming, object-oriented programming, and reusable software," but concludes that these efforts failed to change the fundamentally craft nature of software production (85).[2] "Boys' Toys and Women's Work" (2001) anticipates the recent wave of studies on gender and computing.[3] Mahoney suggests that introducing feminist perspectives could improve technical practice: "Feminist analysis has brought out the ways in which a world built by men hides the ways in which women make it work, and it is the working world that computers must capture if they are going to enhance all our lives" (117).

The final set of essays sketch an intellectual history of computer science. These papers have the feel of works in progress; as Haigh notes in his introduction, they trace chains of intellectual influence but fail to introduce the major actors, contextualize their intellectual endeavors, explain technical concepts, or frame these events within larger questions posed by the history of science. They do provide a meticulously researched account of how a small group of theorists brought together ideas from mathematics, information theory, and linguistics to create a new discipline. One overarching theme is the convergence of the abstract and concrete: "Mathematics provided the structures on which a useful and deep theory of computation could be erected. In turn, theoretical computer science gave physical meaning to some of the most abstract, useless concepts of modern mathematics" (178). While Mahoney acknowledges that some major figures in computer science did not agree that the field was primarily mathematical, their views are relegated to footnotes. Yet he also concedes that the mathematical agenda failed to produce "a coherent general theory" and that "for some questions no mathematics could account in theory for what computing was accomplishing in practice" (156–57). Perhaps it was this impasse that led Mahoney to embrace a more pluralistic notion of "computing(s)."

2. For a recent study of programming as labor, see Nathan Ensmenger, *The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise* (Cambridge, MA: MIT Press, 2010).

3. See David Alan Grier, *When Computers Were Human* (Princeton, NJ: Princeton University Press, 2005); the historical essays in Thomas Misa, ed., *Gender Codes* (Hoboken, NJ: Wiley, 2010); and Janet Abbate, *Recoding Gender* (Cambridge, MA: MIT Press, 2012).

Mahoney's former student Joseph A. November provides a fine example of the "applications turn" within the history of computing. In computer parlance, a "killer app" is one that motivates people to purchase the computer itself. *Biomedical Computing* recounts how the first such applications for biomedicine were developed during the period 1955–1965. Locating agency in both large-scale institutional politics and the agendas of individual researchers, November demonstrates that acceptance of these new tools by researchers and physicians was neither simple nor inevitable.

One of his key arguments is that in the process of harnessing computers to biomedical tasks, both fields were transformed. Biomedicine was considered a largely qualitative field in the 1950s, and therefore an unlikely candidate for computerization. To make it possible to use computers, "many biomedical researchers structured their experiments to include quantitative methods and to exclude the nondigital or nonquantitative types of data computers could not process. To gain access to large, centralized computers, they also integrated—or were pushed to integrate—their centers of research into the infrastructures of both government and corporate power" (8). Conversely, computing machines and practices were reshaped by biomedicine. The small, interactive LINC computer, which was designed to satisfy the preferences of biologists engaged in laboratory experiments, influenced the subsequent development of minicomputers and PCs. DENDRAL, a program to help identify molecules from spectroscopy data, is considered the world's first expert system and spawned a generation of AI research as well as commercial products.

November sometimes seems to equate computing with hardware, giving lengthy descriptions of machines but little attention to how the software that made the applications possible was produced. For example, the chapter on LINC devotes only one paragraph to the development of the LINC programming environment, despite hints in the text that LINC users found the lack of software support to be a major obstacle (180, 184). A notable exception is November's case study of an attempt to computerize medical records at Massachusetts General Hospital in the mid-1960s. He explains how the real-world messiness of medical records, with their endless exceptions and evolving requirements, challenged programmers who were used to the more standardized data of traditional scientific applications. November's analysis sheds new light on the current slow progress in computerizing medical records.

Sherry Turkle's *Simulation and Its Discontents* also locates itself firmly in the applications world. Her subject is the epistemological anxiety that arises from the imposition of computer models between the scientist or designer and the material world they seek to understand. Turkle is well known for her psychological and ethnographic studies of computing, and this volume follows familiar threads: the thrill of embodied human encounters with computers as well as concerns over technology's power to alter the ways we think.[4] Her title essay is followed by four case studies contributed by computer scientist William J. Clancey, architect Yanni A. Loukissas, and anthropologists Stefan Helmreich and Natasha Myers.

The two case studies that provide the data for Turkle's essay were conducted in 1983–1987 and 2002–2005, allowing her to comment on changes and continuities from the early days of simulation to its current ubiquity. Explaining her focus on "discontents," she asserts that these "draw our attention to settings where simulation demands unhappy compliance . . . to things that simulation leaves out" (5). She quotes a professor interviewed in 1984 who insisted that "practitioners must learn 'to do' and 'to doubt,'" immersing themselves in simulations while retaining a critical awareness of what may be distorted or missing (7). Much of the discontent expressed by her informants stems from their impatience with colleagues who, in their view, either accept simulations uncritically or stubbornly refuse to take advantage of them.

The book's greatest strength is the ethnographic richness of the case studies. Loukissas reveals architects' worries that CAD systems will limit their creativity, undermine the traditional master-apprentice relationship, and hand over control of the design process to engineers. Several chapters focus attention on the body. Clancey recounts how members of Mars rover teams experience simulation as embodiment, and Myers describes how scientists use their bodies as tools for understanding protein structures and movements. Helmreich's study of oceanographic submersible robots notes how bodily discomfort, such as seasickness on board a research boat, becomes a badge of authentic engagement with the environment being studied remotely.

While the various essays are compelling, the book as a whole lacks a central argument. The authors never present a shared definition of simulation and what, if anything, sets it apart from other visualization techniques in science. Does simulation imply an immersive virtual reality, like the "fly through" architectural plans described by Loukissas, or could it include any type of computer model? The chapters on undersea and outer space exploration

4. Sherry Turkle, *The Second Self: Computers and the Human Spirit* (New York: Simon & Schuster, 1984); and Sherry Turkle, *Life on the Screen: Identity in the Age of the Internet* (New York: Simon & Schuster, 1995).

describe techniques that could be considered mediation rather than simulation, since they involve remote interactions with real physical environments, while Myers does not even use the word *simulation*, preferring the term *modeling*. Also notable is the absence of the trope of "discontent" from Helmreich's and Myers's chapters, in which "intimate sensing" and ecstatic performance hint at a more transcendent experience of simulation (131).

Perhaps the ultimate "killer app" is one that promises to transform an entire nation. Eden Medina's *Cybernetic Revolutionaries* recounts how Chilean engineers in Salvador Allende's administration (1970–1973) tried to harness cybernetics to bring about a peaceful socialist revolution. Medina's book is a welcome addition to a small but encouraging trend: historical studies of computing outside the United States or Western Europe.[5] She describes how Stafford Beer, the British "father of management cybernetics," joined forces with Allende's engineers to help create Project Cybersyn (19). They envisioned a network of telex machines located at factories and other production sites that would send data to a central computer running a cybernetic model of the Chilean economy. Real-time economic indicators would then be displayed in a futuristic control room to guide government intervention. While Cybersyn did not outlast the violent fall of Allende's government, it is striking that a country with few technical personnel or resources proposed—and at least partially realized—a computerized system far more ambitious than anything the United States had yet achieved. Medina's account "challenges simple models of technological diffusion that frame science and technology as flowing from north to south" and suggests that the unique conditions of less-developed nations can present advantages, not just limitations (9).

More broadly, Medina asserts that "political innovation can lead to technological innovation" by creating a demand for particular tools and by changing the criteria by which technologies are judged (44).[6] She draws out the ways in which "Beer and Popular Unity [Allende's party] were exploring similar intellectual terrain in the different domains of science and politics" (16). These similarities included a goal of creating large, complex systems that were both adaptable and stable; a commitment to democracy and freedom of action for individuals within the system; a belief that a balance between local

5. One such study that makes a good companion piece to Medina's is Slava Gerovitch, *From Newspeak to Cyberspeak: A History of Soviet Cybernetics* (Cambridge, MA: MIT Press, 2004).

6. For another example of politically driven innovation, see Jennifer S. Light, *From Warfare to Welfare: Defense Intellectuals and Urban Problems in Cold War America* (Baltimore, MD: Johns Hopkins University Press, 2005).

autonomy and top-down control could be achieved by creating feedback systems among the various levels; and a desire to apply their ideas to the real world. The engineers charged with building Cybersyn tried to further embed socialist values in the system by designing it to promote egalitarian relations among its users, rather than imposing centralized control and a surveillance regime. Here was utopian computing in its purest form.

Cybersyn could be seen as the realization of a "technomyth," the term proposed by Paul Dourish and Genevieve Bell for the utopian visions motivating much computer science research. In *Divining a Digital Future,* Dourish, a computer scientist, and Bell, a cultural anthropologist, describe the recent rise of a computer science subfield called ubiquitous computing or "ubicomp." Ubicomp researchers envision computer power embedded everywhere in the material world around us, invisibly and effortlessly supplying our thirst for information. The authors locate the origins of this technomyth in a 1991 article by Mark Weiser of Xerox PARC. Weiser argued for a paradigm shift from the desk-bound personal computer to a range of specialized, distributed, networked devices. The authors note how Weiser was influenced by fellow PARC researcher and anthropologist Lucy Suchman, who led him to think about "situational use" in the real world (11). As developed by Weiser and his successors, ubicomp is not a specific technology—though it relies on technologies such as networks and sensors—but rather an approach to creating a particular type of user experience.

Historians of science might wish to see more reflection on what it means for a research area to be defined in this loose way. What kinds of boundary work, training, and funding structures maintain the field's identity? Dourish and Bell seem to take for granted that ubicomp is both desirable and inevitable; have objections or alternatives been suggested? While these questions remain unanswered, subsequent chapters offer a rich ethnographic exploration of "information technology as a site of cultural production" by focusing on themes of infrastructure, mobility, privacy, and domesticity (189). Bell and Dourish argue that ubicomp discourse locates the field in a "proximate future" where ideal, seamless systems will supposedly be achieved, allowing researchers to ignore the "mess"—the many layers of technology, organization, and regulation—that is needed to make systems function in the present (20). "Mess" also invokes the persistent plurality of visions of what this technology is or should be (4–5). Like Medina, the authors question the assumption that U.S. norms of computing should be taken as a standard. For example, they show how privileging the U.S. paradigm of market-led, office-oriented, individualistic technologies marginalizes the achievements of ubicomp projects in Singapore and Korea that are government-led and focus on public transportation, domestic and recreational use, and the collective good (29–39).

A common insight of these books is that before any “killer app” can be introduced, the application area must be transformed to fit the computer. In biomedicine, research and clinical practice had to be quantified. Turkle’s informants had to radically alter laboratory practices, research goals, and student instruction to support simulation. In Chile, production facilities were linked by a communications network, and factory managers would have had to change their operations to provide and respond to real-time feedback. In short, the discipline that gets computerized is not the original discipline, but a computer-ready version with new rules and priorities; practices that do not fit with the new technology are devalued or discarded. Dourish and Bell aim a critical lens at this transformative process, noting that when ubiquitous computing evangelists envision “a future in which our encounters with the world and each other are smoothed by the application of technology,” they ignore the vital messiness of the real world (91).

As a group, the books reviewed here begin to fulfill Mahoney’s pluralistic vision of “histories of computing(s).” Rather than placing the machine at the center, they give careful attention to the ways in which the agendas and myths of communities of users and practitioners shape both the technology and meaning of computing.