When Social Networking Was Social

Michael D. Gordin

Reviews in American History, Volume 48, Number 1, March 2020, pp. 159-164 (Review)

Published by Johns Hopkins University Press
DOI: *https://doi.org/10.1353/rah.2020.0021*

WHEN SOCIAL NETWORKING WAS SOCIAL

Michael D. Gordin

Joy Lisi Rankin, *A People's History of Computing in the United States.* Cambridge, MA: Harvard University Press, 2018. 336 pages. Figures, maps, notes, bibliography, and index. $29.95.

Two transformative technologies emerged out of World War II, and for decades historians—following the world at large—focused on the wrong one.[1] Nuclear fission not only powered the bombs that destroyed Hiroshima and Nagasaki, it promised to either save humanity through cheap atomic energy and medically relevant radioisotopes, or to destroy us all if the U.S.-Soviet Cold War turned hot. The historiography mushroomed like the clouds these weapons released. At first, the work was narrowly technical (how was the bomb made and used?) or political (how was international and/or domestic control to be negotiated?), but with the growth of social and cultural history it came to encompass civil defense, reactor meltdowns, films, protest movements, and much more. We are left with a rich body of scholarship exploring almost every aspect of a technology that was supposed to alchemize everything but ended up leaving intact much of how things were done before scientists split the atom. We never got safe energy too cheap to meter, but we also never got the fiery apocalypse, so we can call it even.

What about that other amazing technology of the war? News about it spread with less fanfare, ironically because it had done a great deal to secure victory for the Allies and disclosure would diminish its future strategic significance. Unlike the atomic bomb, whose cultural power relied upon its being dropped on cities or telegenically detonated before cameras, the digital calculating machine was a placid newborn. Still, these new "computers"—until the end of the war, the term referred to the workers, usually women, who operated calculating machines rather than to the devices—at first made quite a din. The bombes at Bletchley Park and the ENIAC (Electronic Numerical Integrator and Computer, first put to work on 10 December 1945) were massive things, and massively expensive. They crunched numbers, enabling the resolution of thorny numerical calculations crucial for the national security state. Fast forward a couple of decades, and you may well be reading these words not

Reviews in American History 48 (2020) 159–164

on paper but on a descendent of those room-sized behemoths that is orders of magnitude more powerful. Possibly it fits in your pocket.

The historiography chronicling this metamorphosis was rather slower to get off the ground than its nuclear counterpart, partly because of the unwillingness of private businesses to risk leaking trade secrets. Plus, I suspect, historians were waiting to see whether these doohickeys were really going to shake things up or not. The early scholarship focused on the hardware: how did the machines actually work, and how did people figure that out? Unless you are an electrical engineer, these studies made for dry reading, while the flashier emergence of "cybernetics" (1948), morphing into "artificial intelligence" (1956), seemed at first too weird to tackle. The narrative began to pick up in the 1960s, as the United States witnessed the boom in minicomputers, then microcomputers, then "personal computers" in the 1970s. The historical consensus was slow to congeal, but it has proved tenacious.[2] I expect most teenagers with a smartphone can rattle off a variant of the story.

As Joy Lisi Rankin relates in her forcefully revisionist *A People's History of Computing in the United States*, that standard picture—she calls it the "Silicon Valley mythology"—while not entirely inaccurate, is more a wildly successful branding exercise than a true history of how the Atomic Age turned into the Information Age. Her introduction provides an especially clear articulation of the incomplete picture painted by this mythology and what is wrong with it:

> This compelling myth tells us that, once upon a time, modern computers were big (and maybe even bad) mainframes. International Business Machines, much more familiar as IBM, dominated the era when computers were the remote and room-size machines of the military-industrial complex. Then, around 1975, along came the California hobbyists who created personal computers and liberated us from the monolithic mainframes. They were young men in the greater San Francisco Bay Area, and they tinkered in their garages. They started companies: Steve Jobs and Steve Wozniak established Apple; Bill Gates and Paul Allen developed Microsoft. Then, in the 1990s, along came the Internet to connect all of those personal computers, and the people using them. Another round of eccentric nerds (still all young white men)—Jeff Bezos, Sergey Brin, Larry Page, and Mark Zuckerberg among them—gave us Amazon, Google, Facebook, and the fiefdoms of Silicon Valley. (p. 2)

The big problem—aside from the very bro-centric, self-congratulatory tone of this ode to progress—is that the emphasis of the Silicon Valley mythology is on the machines as *commercial products*, and how to sell them to what Rankin calls "computing consumers." This explains some of how we get to the status quo, with every individual increasingly expected to purchase multiple personal devices to take advantage of the computing revolution (or just to order a taxi).

Rankin's book sidesteps this narrative, presenting a largely parallel history that emphasizes "computing citizens." The object she follows is not the minicomputer or microcomputer, as had the stodgier old scholarship or the

updated Geek Epic she points to, but rather the *network*. (There's a related historiographical dragon to slay here, linked to the military-funded ARPANET, an account that sits comfortably next to the nuclear historiography on the military-industrial-academic complex, and Rankin proposes a complement, if not quite an alternative, in chapter 7.) The development that animates her whole story is the "time-sharing" computer that chopped up programs into smaller bits so that instead of the mainframe being monopolized by one user's program at a time ("batch programming"), it allocated processing among multiple users who could interact simultaneously. Time-sharing subverted some of the logic of the big IBM machine—so expensive that to recoup the cost owners needed to line up batch programmers 24 hours per day, like air-traffic control at a major metropolitan airport—and instead enabled social, collaborative computing in a more "decentralized" manner. I put "decentralized" in scare quotes because there was still typically just one centralized computer that processed all the various inputs, but the experience of the users was not of a single machine but of the many terminals connected to it.

Time-sharing was ideal for education. One of Rankin's most significant methodological points is that the historiography has neglected most educational institutions in favor of MIT and Stanford, largely because they fit nicely into the military-industrial or Silicon Valley narratives. Rankin's educational geography is different: Dartmouth College, University of Minnesota, University of Illinois at Urbana-Champaign, and a host of high schools linked to their networks by the telephone grid. Roughly half the book focuses on Dartmouth and the rest on the Midwest, adding to and making visible a diligent cohort of historians of computing who have been working to unearth what happened in the rest of the country. She does not simply incorporate more institutions to the same old story, though. As she notes in a crucial endnote, because many of her actors were self-consciously developing educational tools, "they often included meticulous details of individuals' encounters with the terminal, the keyboard, the screen, the language, the lessons, the appropriate syntax, and similar issues" (p. 246n10). They also documented their work clearly, leaving tracks for future historians. Even though, by the necessity of archival traces, most of Rankin's sources come from institutional repositories, she is able to raise the volume on user experience, especially when she ventures to secondary schools, which are almost completely neglected in conventional accounts.

The first three chapters concentrate on the Dartmouth Time-Sharing System (DTSS) and offshoots from it. Although Rankin strives to work from the bottom up, the account begins from the top down. Frustrated by having to run programs on the MIT mainframe hours away and excited about the potential of digital computing on their small New Hampshire campus, Dartmouth mathematicians Tom Kurtz and John G. Kemeny (the latter of whom would become the president of the college in 1970) arranged to purchase a GE com-

puter in the early 1960s to run an experimental time-sharing system on campus. After fits and starts, the program became a success, and Kemeny in particular devoted enormous efforts to instilling programming across campus. Kemeny and Kurtz co-wrote the flexible programming language BASIC (Beginner's All-purpose Symbolic Instruction Code) to lower the barriers to entry even further, and Rankin dedicates a lengthy and compelling chapter to how the language spread coast-to-coast through the evangelism of enthusiasts.

Rankin uses Dartmouth to illustrate several key themes of the later Midwestern chapters. DTSS was explicitly conceptualized as a *network*, both on campus and linking other universities and high schools that dialed in to program the computer for the cost of a phone call. Long-distance charges piled up quickly, so the geographic limitation mostly to upper New England was an economic constraint first and foremost. The college students, computer technical support (several of whom were women), and high-school teachers and students prioritized openness and sharing, so programs—frequently games, which is important—spread across the network. These are Rankin's computing citizens, experiencing liberation through sharing. Time-sharing as a concept was new enough that finding the appropriate metaphor was elusive. Computer lab habitués would laugh at those who thought of their teletype keyboard and printer as "the computer" (actually located in the basement), but Kemeny noted that there was sense to this misunderstanding. What, after all, is "a telephone," what we now call a landline? Is it the handset plugged into the wall, as most people would say, or is it the network of wires and switchboards across the country? In a sense, both are correct. Kemeny's favored analogy was decidedly collegiate: computing was like a university library. It was the obligation of the institution to pay for the upkeep of the system, and everyone on campus could share in the volumes it stored. This is a rather different vision than Silicon Valley's personal computers, which would have us all buy the books and keep them for ourselves.

At the same time, Rankin sees limitations to the campus-wide embrace of computing in Hanover. Introductory mathematics classes required students to write programs, which framed the computer as fundamentally a mathematical instrument. Rankin speculates that this narrowed the vision of the machine for many more humanistically inclined students. She devotes much more space to a discussion of how Dartmouth's all-male (women were grudgingly admitted in a piecemeal fashion in 1972) and overwhelmingly white student body shaped the culture of computing, a theme she continues in later chapters for different systems. Fraternity life and football were the dominant entertainments on this rural campus, and Rankin explores newsletters, programming notes, reminiscences, and other ephemera to argue that a masculinist (and racially coded) ethos attached itself to the DTSS. There is no question that the students played lots of sports games on the computer and tried using it to

impress their dates, and there is even less question that, as historians such as Nathan Ensmenger (*The Computer Boys Take Over: Computers, Programmers, and the Politics of Technical Expertise*, 2010) and Mar Hicks (*Programmed Inequality: How Britain Discarded Women Technologists and Lost Its Edge in Computing*, 2017) have demonstrated, computing culture became strongly gendered male from an originally more fluid situation, yet Rankin's specific arguments connecting the two occasionally come across as forced. She is right that the moment of transition was the 1960s and that locus was student culture, but the evidence presented here does not fully elucidate the process.

After leaving Dartmouth, Rankin devotes a chapter to a survey of the nationwide interest in the idea of a "computer utility," this time analogizing time-sharing to the electric-power grid. She moves on in chapter 5 to the Minnesota Educational Computing Consortium (MECC) of the mid-1970s, which birthed not only the astonishingly successful *Oregon Trail* video game but also an ambitious experiment to connect public education at all levels with a state-led computing initiative. The geography here is no coincidence; Minnesota was rather futuristic in the 1970s, as described in Thomas J. Misa's *Digital State: The Story of Minnesota's Computing Industry* (2013), which Rankin supplements with archival discoveries. Finally, she devotes two chapters to Donald Bitzer's touch-screen-enabled time-sharing system, adapted for education with the important contributions of his wife Maryann Bitzer, at the University of Illinois: PLATO (Programmed Logic for Automatic Teaching Operations). The name advertises the educational mission from the start, and Rankin weaves in wonderful stories of how the system developed and was used in practice.

As the book's title indicates, Rankin is inspired by Howard Zinn's *A People's History of the United States* (1980). Zinn wanted to tell a social history of the United States that emphasized those voices usually excluded: Native Americans, African Americans, women. Rankin strives for some of the same with respect to computing. We meet high school students, casual programmers, minor bureaucrats, women technicians, African American scholarship students, and others who do not figure in the Silicon Valley mythology. There are limits to how far she can go, of course, because even time-sharing required expensive mainframes and access to some educational infrastructure, and her sources naturally push her to the campuses of major institutions of higher learning. It is nonetheless a significant improvement to the historiography, and will help build a better integrated picture in the future.

That new history, however, will have to move beyond her formulation of "computing citizens." In reaction to the Silicon Valley mythology, Rankin insists that these are "citizens" rather than "computing consumers" because with time-sharing they actually helped build the programs and even the operating systems of these nascent time-sharing networks. The difficulty comes with thinking of the categories as mutually exclusive: "The thousands of students

and educators using PLATO did not have to pay for their access; they were computing citizens, not computer consumers" (p. 167). As an American citizen, I regularly pay for access to government services; this is the primary function of taxes. One does not have to be a Marxist to suspect that some aspects of my American citizenship are not unrelated to my also being an American consumer. Rankin often sharpens the blurry line between citizenship and consumership. To stick with the PLATO example, she tracks its inspiring educational career, and notes in passing that Bitzer's firm, Control Data Corporation, attempted to commercialize time-sharing as well. It fizzled, much like Western Union's contemporary efforts to add time-sharing to their money-order and telegraph services. Those stories are well chronicled in an unpublished dissertation by Christopher McDonald, which Rankin cites but does not substantially engage.[3]

The historiography of the "rise of computing"—or whatever we want to call this swath of literature—has in many ways recapitulated the developments in the nuclear historiography that overshadowed it for decades. Hypertechnical scholarship was followed by a political turn (complicated by the interaction with the policy literature that evolves alongside the historiography), and Rankin's *A People's History of Computing* contributes to an emerging wave of social and cultural history. The strength of this literature is the local texture, visible here in abundance, but future scholarship needs to push further in the direction of integrating those cultural pictures with the business history of how we lost this alternative history of cooperative computing and instead became thralls to an oligopoly of a half dozen technology firms. Rankin offers a whirlwind account in the epilogue of her book, which sketches out some of the pathways that future scholars would be well-advised to widen and deepen. Collectively, we need yet more research of this quality to understand how our current model of "personal computing" so displaced another kind of social computing to such a degree that even the Internet could not bring back Kemeny's dreams of computing as a futuristic lending library.

Michael D. Gordin is Rosengarten Professor of Modern and Contemporary History at Princeton University, and Chief Research Fellow at the Poletayev Institute for Theoretical and Historical Studies in the Humanities, National Research University Higher School of Economics, Moscow. His most recent books are *Einstein in Bohemia* and, co-edited with G. John Ikenberry, *The Age of Hiroshima*, both published by Princeton University Press in 2020. This article was prepared within the framework of the HSE University Basic Research Program and funded by the Russian Academic Excellence Project "5–100."

1. I owe this observation to my late colleague, historian of computing Michael S. Mahoney, who often said this to me, a historian of nuclear weapons.

2. A central scholarly version is Martin Campbell-Kelly and William Aspray, *Computer: A History of the Information Machine*, 2nd. ed. (2004 [1996]).

3. Christopher McDonald, "Building the Information Society: A History of Computing as a Mass Medium," PhD diss., Princeton University 2011.