PROJECT MUSE®

Revolution and Resistance: Rethinking Power in Computing History

Rebecca Slayton, Nathan Ensmenger

IEEE Annals of the History of Computing, Volume 30, Number 1,
January-March 2008, pp. 96-97 (Article)

Published by IEEE Computer Society

➔ For additional information about this article

<https://muse.jhu.edu/article/235252>

Revolution and Resistance: Rethinking Power in Computing History

Rebecca Slayton
Stanford University

Editor: Nathan Ensmenger

If one theme pervades popular imagination and memory about the computer, it is revolution. But the term begs scrutiny. Sometimes the “computer revolution” is linked ironically to the dramatic enhancement of dominant power structures, other times to a radical inversion of social order.¹ Historically, the relationships between transformations in computing and transformations in power have been diverse, complex, and ambivalent. Although nuclear anxieties and defense spending reshaped computing during the Cold War, many leaders in the field began working against military institutions and joined the peace movement by the late 1960s. Women carved out careers for themselves in computing decades before feminist movements hit the streets in the 1970s, but today they are significantly underrepresented in their field and their contributions are often written out of history. And computing has become a key tool of governance, even as many enthusiasts continue to espouse digital technology as a means for challenging dominant government institutions around the world.

These are just a few puzzles suggesting a need for more historical attention to the diverse ways in which computing—as a field of technology, knowledge, and labor—and its broader sociopolitical milieu have shaped one another. Here I will suggest that we might usefully analyze change within and around computing as we would a political movement. Rather than asking primarily how and why technology changed, we may ask who persuaded others of the need for change, how collective identities and ideologies were created or contested in this process, and how computing has provided novel tactics for challenging dominant power structures.

The politics of problem framing

Social movements do not spring up in response to self-evident problems; rather, they are mobilized by charismatic leaders who persuasively define problems for action. Historians have begun to show that a similar dynamic drove the widespread adoption of computers in the latter half of the 20th century: enthusiasts succeeded in defining issues as problems to be solved through computer technology or expertise. But what makes the computer a compelling solution, and for whom? What happens when contention over problem framing erupts not only between computer experts and initiates, but within expert communities?

Problem framing can be a tremendously divisive process, and we often find social and political commit-

ments at the heart of conflict. Consider, for example, the much-discussed software “crisis” that experts began to acknowledge in the 1960s. Everyone agreed that software development tended to overrun time and cost projections, and that glitches were frustratingly persistent. But was this a “crisis,” and if so, for whom? Some experts at the first “software engineering” conference in 1968 expressed concern about “what might happen directly as a result of failure in an automated air traffic control system,” and “worried that our abilities as software designers and producers have been oversold.” But another objected: “As someone who flies in airplanes ... I’m concerned personally about the possibility of a calamity, but I’m more concerned about the effects of software fiascos on the overall health of the industry.”² He was among those who preferred to frame the software problem as a challenge for cost-effective project management rather than as a crisis of reliability. In other words, concerns about professional status and profits shaped the ways that experts framed the woes of large and complex software systems.

Collective identities and ideologies

“Social” or “technical” problems are defined in relation to collective ideologies and identities that can change over time. Historians are just beginning to examine how political visions have shaped computing communities. For example, Fred Turner shows how libertarian ideals shaped the development of early virtual communities in the San Francisco bay area.³ Such studies are tremendously valuable, but also too few.

For example, how have conceptions of the appropriate relationship between computing communities and public politics shaped efforts to establish computing as a unified science or profession? This issue came to the fore in 1969, when a group of computer experts organized a “Counter-Conference” in protest of the imminent Association of Computing Machinery (ACM) conference in Chicago (a site which many found objectionable due to police brutality against the New Left at the 1968 Democratic National Convention). Amid more general anxieties about professionalism in computing, Counter-Conference organizers outlined three defining characteristics of a professional, but first and foremost was the professional’s “acceptance of responsibility for the consequences of his work.”⁴ They demanded more explicit organizational engagement with the “social implications” of computing. By contrast, others felt that

explicit organizational engagement with social issues would undermine professionalism by dividing the community, or by compromising “technical” objectivity.

Many of the counter-conference organizers were offended by the entanglement of computing with the military-industrial complex, and so they aimed to rock the boat. A few solicited views on social responsibility from all nominees for ACM office and promised to publicize the results before the upcoming, very competitive, election. Through these and related activities they sought to establish a “professional” identity apart from the military-industrial establishment.⁴

The professional voice of computing continues to tangle with contentious politics today. For example, when the US government launched its Total Information Awareness program in 2003, the ACM’s US Public Policy Committee objected that it endangered civil liberties.⁵ But the ACM Special Interest Group on Knowledge Discovery and Data Mining soon countered that the funds available through Total Information Awareness would support research on privacy-enhancing technologies.⁶ Who speaks for computing and what social visions do they espouse?

Resistance: Tactics and strategies

Any social movement must choose tactics and strategies for change, and we might usefully analyze the ways in which computing has provided novel means for resisting otherwise hegemonic power. Whereas explicit claims to revolution have often ironically aimed at enhancing dominant powers, very real political challenges have emerged quite tacitly from within computing.

As a new technology and field of expertise, computing provided avenues for challenging power structures. For example, Janet Abbate has described how women’s possession of specialized programming knowledge enabled them to bargain for (somewhat) more equitable treatment in the workplace.⁷ Similarly, Nathan Ensmenger has shown how the novelty of software gave programmers an avenue by which to challenge traditional organizational politics and structures.⁸ Historians highlighting the diverse interests of computer users implicitly highlight the ways in which technology has provided unexpected avenues for political challenges. For example, by the 1980s, the internet that was originally de-

signed for the defense department became a tool for the antinuclear movement.

In short, for groups carving out a novel field of labor, technology, and expertise, computing provided tactics and strategies with which to challenge dominant power structures. How, when, and why have these tactics been effective? How should we understand failure? As presidential candidates take to YouTube and revolutionary movements infiltrate the Web, we could use more historical attention to such questions.

In sum, scholarship on social movements shows how defining problems, forming collective identities, and choosing strategies are crucial means of achieving political change. We have evidence that the same activities have shaped modern computing. To understand the place of power in computing history, we would do well to confront revolutionary claims with revolutionary analysis.

References and notes

1. L. Winner, “Mythinformation,” Chapter 6 in *The Whale and the Reactor: A Search for Limits in an Age of High Technology*, Univ. of Chicago Press, 1986.
2. P. Naur and B. Randell, “Software Engineering: Report on a Conference Sponsored by the NATO Science Committee,” Scientific Affairs Division, NATO, 1969, p. 121; <http://homepages.cs.ncl.ac.uk/brian.randell/NATO/nato1968.PDF>.
3. F. Turner, *From Counterculture to Cyberculture*, Univ. of Chicago Press, 2006.
4. Files of Daniel D. McCracken, on loan to the author.
5. ACM Public Policy Committee to Senate Committee on Armed Services, 23 Jan. 2003; http://usacm.acm.org/usacm/Letters/tia_final.html.
6. “‘Data Mining’ Is NOT Against Civil Liberties”; <http://www.sigkdd.org/civil-liberties.pdf>.
7. J. Abbate, “Proto-feminism and Programming: Gender Politics in Computing Before the Civil Rights Era,” paper presented at the Society for the History of Technology Annual Meeting, 2007, Washington, D.C.
8. N.J. Ensmenger, “Letting the ‘Computer Boys’ Take Over: Technology and the Politics of Organizational Transformation,” *Int’l Rev. Social History*, vol. 48, 2003, pp. 153-180.

Contact Rebecca Slayton at rslayton@stanford.edu.

Contact department editor Nathan Ensmenger at annals-thinkpiece@computer.org.