From "Impact" to Social Process:

Computers in Society and Culture

Chapter 12 of Sheila Jasanoffet al., eds.,
Handbook of Science and Technology Studies
(Beverly Hills, CA: Sage Publications, 1994)

by

Paul N. Edwards
School of Information
301D West Hall
University of Michigan
550 East University
Ann Arbor, MI 48109-1092
pne@umich.edu
http://www.si.umich.edu/~pne/

A couple of years ago I received from a publisher, unsolicited, a copy of a new textbook on computers and social issues. It was a sleek large-format paperback, with a beautifully designed computer graphic on the cover. In imposing black type, the title read: *Computers and Society — IMPACT!*

The sensationalism of this title, with its billiard-ball imagery, nicely encapsulates what is probably the most common view of the relationship between information technology and the social world. Computers are arguably among the half-dozen most important post-WWII technologies, an impressive list which might include television, jet aircraft, satellites, missiles, atomic weapons, and genetic engineering. The proliferation of cheap, powerful information processing and computerized control systems has unquestionably altered — and in some cases deeply transformed — the nature of warfare, communications, science, offices, factories, government, and certain cultural forms. This point hardly requires substantiation; reportage on the "information revolution" has become a virtual cottage industry.[1]

But the exact nature of these "impacts" of computing, as well as the details of how computers are supposed to produce them, remain in dispute. The utopian / dystopian character of much of the analysis in this area is aggravated by its generally ahistorical character. The basis for claims of "impacts" lies more often in broad economic or cultural analysis than in the detailed exploration of local effects characteristic of some of the best science studies literature (Dertouzos 1991; Garson 1988; Roszak 1986; Weizenbaum 1976).

This chapter explores some of the significant social effects of digital computers and some of the social forces shaping their development. Because even a cursory overview of such an immense arena is beyond available space limits, the chapter focuses on three cases: military relations with computing in the post-WWII era, the "productivity puzzle" of computerization in banking, and the relationship of gender identity to computer use. The essay has two goals. First, it offers the uninitiated a point of entry into some of the vast literature on computers and society. Second and more importantly, it provides a historical and social analysis that treats computers not merely as causes but also as effects of social trends.

In this I take as a given that technological change is, as Merritt Roe Smith has put it, a *social process:* technologies can and do have "social impacts," but they are simultaneously social *products* which embody power relationships and social goals and structures (Smith 1985). Social impacts and social production of artifacts in practice occur in a tightly knit cycle. The three cases presented here show how any full-blooded analysis must reflect the complexity of this interaction.

I. Computers and the Military after World War II

The US armed forces have been the single most important source of support for advanced computer research from World War II to the present. How did this support affect the technology itself? How did the new technology affect military doctrines and institutional structures? The historical analysis presented here demonstrates how military needs and priorities guided computer development, especially in its first two decades, and shows how computers, in turn, shaped the military.

Historians now generally recognize John Atanasoff of Bell Laboratories as the inventor, in 1940, of the first electronic digital computer. But while Atanasoff and others created this and other prototypes just before the US entered World War II, their significance went for the most part unrecognized. This was largely because *analog* computers, such as the differential analyzer of Vannevar Bush, were already a well-developed technology.[2] Bush built a series of these machines, which were highly though not perfectly accurate for solving complex differential equations, culminating in one built in 1942 at MIT which was fully programmable using punched paper tape (Goldstine 1972, pp. 92-102 *et passim*). New analog computers, such as those used in antiaircraft weapons, were among the decisive technical achievements of the war (Fagen 1978). But the feverish technical developments of WWII weaponry generated demand for huge numbers of computations to solve ballistics and coding problems — and, because of their urgency, for unprecedented rates of speed. It was to this end that programmable, electronic digital computers, capable of dramatically faster calculation, were developed.

The first of these were created by US and British military forces. The Electronic Numerical Integrator and Calculator (ENIAC) was constructed at the Moore School of Engineering in Philadelphia between 1943 and 1946 by the US Army Ordnance Department. Its purpose was to automate the tedious calculation of ballistics tables, on which antiaircraft weapons and artillery then depended for accuracy. During the war these calculations were performed by a mostly female corps of young mathematicians, known as "computers," using hand calculators. When the ENIAC project began, some of these women became its first programmers — hence the sobriquet "computer" for the new machine. The ENIAC was not completed until after the war's end, when it was immediately put to work on physics equations connected with thermonuclear weapons for the Los Alamos laboratories. (It failed to solve some of them, producing demands for more powerful machines.) Among the many influential members of the ENIAC development team were John von Neumann, who developed the serial control architecture which now bears his name, and J. Presper Eckert and John Mauchly, who proposed and directed the project and were responsible for most of the ENIAC's key design features. Eckert and Mauchly started their own company — UNIVAC, the first commercial computer producer — in

1946 using knowledge gained from working on the ENIAC and its successor, the EDVAC.[3]

Credit for the first operational electronic digital machine, however, belongs to the British "Colossus," constructed at Bletchley Park with the participation of Alan Turing, the mathematician who had invented the theory of digital computation in 1936 (Turing 1936). The first Colossus was completed in 1943 and used throughout the rest of the war to break the Enigma and Fish ciphers used by the German high command. The machine's great speed and accuracy, compared with existing hand calculation techniques and automated analog computation, enabled it to crack the cipher quickly enough for intercepted messages to be useful to the Allies. The Colossus thus played a major— perhaps even a decisive — role in preventing Britain's defeat and assuring a subsequent Allied victory (Hodges 1983).

WWII-era computer development, then, may be characterized as *need-driven research*. Ideas for automating calculation came from scientists and engineers. They were adopted by the military because of specific, pre-existing needs for calculation. WWII-era computers produced only limited impacts on the military, since they were used simply to speed up existing processes. But these military projects did produce local concentrations of researchers working on electronic digital techniques, and these groups persisted after the war, providing the social and organizational nucleus for future research. At this point, computers were clearly more a social product than a driver of social change.

Computer development in the 1945-55 period occurred very rapidly, with projects such as the National Bureau of Standards SEAC, von Neumann's Institute for Advanced Studies (IAS) machine and its several copies, and Eckert and Mauchly's BINAC (built as a guidance computer for Northrop's Snark missile). Almost every new machine incorporated new innovations. The UNIVAC team struggled to create and introduce a production computer (it finally succeeded in 1951 and subsequently sold 46 UNIVAC I's), but most machines were one-of-a-kind, experimental prototypes. Then as now, technical advancement occurred with astonishing speed. Indeed, statistical measures of computer development, such as the rate of doubling of random access memory capacity and the halving of cost per computation, became and remain virtual tropes of progress and technological "revolution" (see Figures 1 and 2, below).

Perhaps bedazzled by this muscular technical progress, most historiography of computing has focused on three things: (1) the technical characteristics of devices, (2) the biographies of individuals responsible for important innovations, and (3) the intellectual history of computing as a problem of mathematics and engineering (Mahoney 1988). Until recently (Flamm 1987; Flamm 1988; Noble 1984; Winograd 1991) few historians had much to say about the social relations involved in computer R&D — in particular, the meaning of military sponsorship.

Did military needs influence computer technology after WWII, when the wartime research laboratories were dissolved or returned to civilian control? The United States' new status as a superpower, the central role of science and technology in the war effort, the massive wartime federal funding and the associated advancement of communal aims for science, and other factors all contributed to the emergence of a powerful scientists' lobby for continued federal sponsorship, on the one hand, and a wholly new sense within the armed forces of the importance of science and technology — and the potential contribution of "civilian" scientists and engineers — on the other. The incipient Cold War was the final element which allowed military organizations, especially the Office of Naval Research (ONR), to become the default federal sponsors of science and technology R&D in the 1940s and 50s (Dickson 1984; Forman 1987; Edwards 1989; Edwards forthcoming; Smith 1991). Still, most computer R&D projects took place not in military facilities, but in industrial or university laboratories. This was consistent with the general pattern of postwar federal sponsorship of science and technology (Smith 1991). Since so many areas of science and technology benefited from the ONR's relatively non-directive funding, many historians have neglected military influences because of the idea that "everyone was feeding from the same trough."

But military sponsors did not need to undertake detailed direction of research projects in order to achieve their goals, which were in any case of a very general character in relation to new technologies such as the computer. They could rely, instead, on the mere requirement of a plausible military justification for research projects (Winograd 1991). The civilian scientists' and engineers' own imaginations, combined with their wartime experience of military research problems, generated new military ideas in large numbers. These frequently proved far more ambitious and farsighted than those of the military's own leaders, wrapped up in a military traditionalism rendered problematic by new technologies of war (Gray 1991; Gray 1989).

At least in the computer field, a process of *mutual orientation* occurred, in which engineers constructed visions of military uses of computers in order to justify grant applications, while military agencies directed the attention of engineers to specific practical problems computers might resolve.

The most sophisticated leaders, both military and civilian, had an explicit understanding of this mode of *enrolment* of civilian scientists, engineers, and other intellectuals (Callon 1987; Latour 1987).Vannevar Bush, for example, in his famous report on postwar science policy, *Science: The Endless Frontier*, cited the Secretaries of War and the Navy to the effect that

> This war emphasizes three facts of supreme importance to national security: (1) Powerful new tactics of defense and offense are developed around new weapons created by scientific and engineering research... (3) war is increasingly total war, in which the armed services must be supplemented by active participation of every element of civilian population. To insure continued preparedness along farsighted technical lines, the research scientists of the country must be called upon to continued in peacetime some substantial portion of those types of contribution to national security which they have made so effectively during the stress of the present war (Bush 1945, p. 12).

Another indirect channel for military influences on technology was the marketplace itself. The sheer size of the increasingly high-technology armed forces ensured corporate investment in military-related R&D projects. The development of the transistor — privately financed by Bell Laboratories, but with military markets its major rationale — is the best-known example. But there are others of equal importance. The DoD sponsored the development of integrated circuits in the 1950s and purchased the *entire* first-year output of the integrated circuit manufacturing industry, worth $4 million, mostly for use in Minuteman nuclear missile guidance systems. Two major programming languages, COBOL (in the 1960s) and Ada (in the 1980s), were products of standard-setting efforts initiated by the military to assure software compatibility among different projects. Military sponsorship of and specifications for very-high-speed integrated circuit (VHSIC) fabrication in the 1980s led to initial American leadership in the field — followed by failures due to poor cost performance of equipment designed for the military's "high-spec," small-lot production needs (Flamm 1988, 1987; Brueckner and Borrus unpublished ms.; Jacky unpublished ms.; Rosenberg 1986; Winograd 1991).

Military influences on computer technology were thus widespread, but were frequently the product of *indirect* forms of intervention that go unnoticed in traditional historical analysis.

Project Whirlwind and the SAGE air defense system

Probably the single most important computer project of the decade 1946-56 was MIT's Whirlwind. Whirlwind, under the direction of engineer Jay Forrester, actually began in 1944 as an *analog* computer for use in a flight simulator, funded by the Navy. News about the ENIAC and EDVAC digital computer projects led Forrester to abandon the analog approach in early 1946. But the original application goal of a flight simulator remained. Flight simulators of the day were servo-operated, mechanical imitations of airplane cockpits which simulated an airplane's attitudinal changes in response to its controls, giving novice pilots a chance to practice without the expense or the risk of actual flight. In theory flight simulators were, and remain, what is known as a “dual-use” technology, equally useful for training military and civilian pilots. But the urgency of the WWII air war made them *in practice,* in 1940-45, a military technology. This practical goal distinguished Whirlwind from almost all other digital computer projects of this era, because it required a computer which could (a) be used as a control mechanism, and (b) could perform this function in real time.

It is important to emphasize that at this historical juncture these were not obvious goals for a digital computer.

- Analog computers and control mechanisms (servomechanisms) were well-developed, with sophisticated theoretical underpinnings. (Indeed, Forrester began his work at MIT as a graduate student in Gordon Brown’s Servomechanisms Laboratory.)
- Analog controllers did not require the then-complex additional step of converting sensor readings into numerical form and control instructions into waveforms or other analog signals (Valley 1985).
- Mechanical or electro-mechanical devices were inherently slower than electronic ones, but there was no inherent reason why electronic computers or controllers should be *digital,* since many electronic components have analog properties. Numerous electronic analog computers were built during and after the war.
- Most other projects saw electronic digital computers as essentially giant calculators, primarily useful for scientific computation. Their size, their expense, and this vision of their function led many to believe that once perfected only a few — perhaps only a couple — of digital computers would ever be needed. Even Forrester at one time apparently thought that the entire country would eventually be served by a single gigantic computer (Brown 3/15/73).

The technology of digital computation had not yet achieved what Pinch and Bijker call “closure,” or that state of technical development and social acceptance in which large constituencies generally agree on its purpose, meaning, and physical form (Pinch and Bijker 1987). The shape of computers, as tools, was still extremely malleable, and their capacities remained to be envisioned, proven, and established in practice.

By 1948, with its interest in a super-sophisticated and by now extremely expensive flight simulator rapidly declining, the ONR began to demand immediate, useful results in return for continued funding. This dissatisfaction was largely due to Whirlwind's truly colossal expense. Where the cost range of computers like the UNIVAC lay typically between $300 and $600 thousand current dollars, the Whirlwind group was planning to spend a minimum of *$4 million*. "...MIT's funding requests for Whirlwind for fiscal 1949, almost $1.5 million, amounted to roughly 80 percent of the 1949 ONR budget for mathematics research, and about 10 percent of the *entire* ONR budget for contract research" (Flamm 1988, p. 54). The actual budget for that year was $1.2 million — still an amazing level of investment, by any standard, in a single project.

Whirlwind's "estimated completion costs... were about 27 percent of the total ... cost of the entire DoD computer program" (Redmond and Smith 1980, p. 154). By March, 1950 the ONR had cut the Whirlwind budget for the following fiscal year to only $250 thousand. Compared with the $5.8 million *annual* budget Forrester had at one point suggested as a comfortable figure for an MIT computer research program including military and other control applications, this sum was virtually microscopic.

Forrester therefore began to cast about for a new institutional sponsor — and for a new military justification. He was in a special position to do this for a number of reasons. First, Forrester's laboratory entertained a steady stream of visitors from both industry and military centers, each with questions and ideas about how a machine like the Whirlwind might be used to automate their operations. Forrester's notebooks indicate that between 1946 and 1948 these visitors raised dozens of possibilities, including military logistics planning, air traffic control, damage control, life insurance, missile testing and guidance, and early warning systems (Forrester 1946-48). Second, Forrester "shared the apprehensions of Navy Special Devices Center (SDC) personnel regarding confidential projections of a Russian atomic strike capability by 1953" and believed his work could make a personal contribution (Redmond and Smith 1980, p. 150).

Finally, Forrester and his group had been deeply concerned with the issue of military applications all along. In early 1946, when Forrester first reported to the Navy on his plan to switch to digital techniques, he had included several pages on military possibilities. "In tactical use it would replace the analog computer then used in 'offensive and defensive fire control' systems, and furthermore, it would make possible a 'coordinated Combat Information Center,' possessing 'automatic defensive' capabilities, an essential factor in 'rocket and guided missile warfare'" (Redmond and Smith 1980, p. 42, citing Forrester). In October 1947, Forrester, SDC leader Perry Crawford, and Whirlwind co-leader Robert Everett had published two technical reports on how a digital computer might be used in anti-submarine warfare and in coordinating a naval task force of submarines, ships, and aircraft. That year, in frequent meetings at its Sands Point headquarters, Crawford and other SDC personnel had encouraged Forrester and Everett "to see more ambitious prospects of the sort that had stimulated the forward-looking systems-control views represented by their L-1 and L-2 reports" (Redmond and Smith 1980, p. 120).

The following year, as continuation of ONR support became increasingly uncertain, MIT president Karl Compton requested from Whirlwind a report on the future of digital computers in the military. The group produced a

> sweeping vision of military applications of computers to command and control tasks, including air traffic control, fire and combat control, and missile guidance, as well as to scientific calculations and logistics. The estimated cost of this program was put at $2 billion [current dollars], over 15 years. The ... flight simulator [project] was replaced by the broader concept of a computerized real-time command and control system (Flamm 1988, pp. 54-55).

Indeed, the report discussed most of the areas where computers have eventually been applied to military problems (Forrester 1948).

Finally, working with the so-called "Valley Committee" (headed by another MIT professor, George E. Valley), Forrester constructed a grand strategic concept of national perimeter air defense controlled by central digital computers (Jacobs 1983; Redmond and Smith 1980). These would monitor distant-early-warning polar radars and, in the event of a Soviet bomber attack, automatically assign interceptors to each incoming plane, direct their flight paths, and coordinate the defensive response.

Military research budgets took a steep upward turn as a result of the Soviet explosion of an atomic bomb in 1949 and the outbreak of war on the Korean peninsula in 1950 (Forman 1987). By that time, because of its control of nuclear weapons, the Air Force had emerged as the military focus of the Cold War, the most forward-looking and technologically oriented of the armed services. In 1950 the Air Force took over Whirlwind's support from the ONR. Under Air Force sponsorship, the Valley Committee plan rapidly evolved into the SAGE (Semi-Automated Ground Environment) air defense project.

However, the Air Force's primary commitments were to *offensive* strategic forces. Commanders at the highest levels believed that an effective defense against a full-scale Soviet nuclear attack — even without missiles — was a virtual impossibility. They preferred to rely on a policy of "prompt use" of nuclear weapons, a euphemism for pre-emptive strike (Herken 1983). Under this strategy, air defense would naturally be unnecessary. Forrester's group was ridiculed as "the Maginot Line boys from MIT." General Hoyt Vandenberg called the project "wishful thinking" and noted that

> ...the hope has appeared in some quarters that the vastness of the atmosphere can in a miraculous way be sealed off with an automatic defense based upon the wizardry of electronics... I have often wished that all preparations for war could be safely confined to the making of a shield which could somehow ward off all blows and leave an enemy exhausted. But in all the long history of warfare this has never been possible (General Hoyt S. Vandenberg, cited in Schaffel 1989, p. 15).

The Air Force especially feared that emphasis on air defense would reduce budgets for the nuclear-offensive Strategic Air Corps (SAC). But it was essentially forced by political pressures to produce something that looked like an active air defense in order to assuage public fears of nuclear attack. These fears, combined with the "can-do" technological mindset of the MIT engineers, generated the momentum necessary for the SAGE project. Eisenhower ended up supporting both the SAC and the continental air defense program under his high-technology New Look defense strategy.

Valley's group quickly became convinced of the effectiveness of Forrester's digital techniques. But the digital approach involved a major restructuring of Air Force command systems, since it was centralized and automated rather than decentralized and pilot-oriented. A competing project at the University of Michigan, based on analog technology, would have retained the basic command structures but speeded up the calculation process with analog computers. The Air Force continued to fund the Michigan project until 1953. Even then, the Air Force only canceled the project when MIT threatened to quit if it did not commit to the digital approach.

The first SAGE sector became operational in 1958. Its control center consisted of a windowless four-story building with six-foot-thick blast-resistant concrete walls. The Whirlwind machine became the prototype for its contents, the FSQ-7 production computer, manufactured by IBM. "Composed of seventy cabinets filled with 58,000 vacuum tubes, the FSQ-7 weighed three hundred tons and occupied 20,000 square feet of floor space, with another 20,000 square feet devoted to display consoles and telephone equipment." By 1961 all 23 sectors were working. The total cost of the project in the 1950s was somewhere between $4 and $12 billion. Parts of the system operated — using the original vacuum-tube computers — until the mid-1980s (Jacobs 1983).

Whirlwind and SAGE were responsible for many, many major technical advances. The list includes the invention of magnetic core storage, video displays, light guns, graphic display techniques, the first algebraic computer language, and multiprocessing. Many of these advances bear the direct imprint of the military goals of the SAGE project and the political environment of the postwar era — another example of the social shaping of technology. I will mention just three examples.

First, as Paul Bracken has pointed out, the Cold War, nuclear-era requirement that military systems remain on alert twenty-four hours a day for years on end represented a completely unprecedented challenge not only to human organizations, but to equipment (Bracken 1984). The Whirlwind computer was specifically designed for the extreme reliability required under these conditions. It was the first duplexed computer (i.e., it was actually two computers running in tandem, one of which could take over from the other on the fly in case of failure). For the same reason, the machine had a fault-tolerant architecture and pioneered methods of locating component failures. Whirlwind research also focused heavily, and successfully, on increasing vacuum tube lifespan, a major cause of breakdowns in early computers. Down time for FSQ-7 machines was measured in minutes per year — other computers of that era were frequently down for numbers of *weeks*.

Second, SAGE was the first large control system to utilize a digital computer. It translated radar data into fighter-interception coordinates and flight paths, relayed to pilots by radio. Real-time operation was a demand imposed by the control function of the SAGE system. This required, first, much faster operating speeds than any other machine of that period, not only for the central processing units but for input and output devices as well. Second, it required the development of methods of interconverting sensor and control signals from analog to digital form. For example, radar signals were converted to digital impulses for transmission over telephone lines.

Finally, this long distance digital communication was used both for transmission of data from radars and for coordination of the SAGE centers. SAGE was thus the first computer network, a requirement of the centralized command structure. But this centralization was itself a product of SAGE. It was both a technological impact, since without its high-speed communication and coordination, central control on such a scale would not have been possible, and a social product, since SAGE was envisioned by “system builders,” in Thomas Hughes’ phrase, who constructed technologies to fit a visionary ideal (Hughes 1987).

How do the Whirlwind and SAGE projects exemplify social process in the history of computers? Three important points may be made.

First, considered as a politico-military venture, the value of the SAGE project — like its 1980s counterpart, the “Star Wars” strategic defense system — was almost entirely imaginary and ideological. Its *military* potential was minimal, but it helped create a sense of active defense that assuaged some of the helpless passivity of nuclear fear. Civilian political leaders, the incipient corps of military technocrats, and engineers with an almost instinctive belief in technological solutions for politico-military problems — all riding on the technological successes of WWII — thus allied against the Air Force around an essentially ideological program of technological defense. Real-time control computers were a product of these social forces.

Second, in discussions of military contracts, it is common to dismiss "grantsmanship," or the deliberate tailoring of grant proposals to the particular aims of funding agencies, as insignificant to research outcomes. Supposedly, grant proposals that justify basic research in terms of applications are simply a vehicle to obtain funds which both recipients and agencies know will really be used for something else.

In the case of Whirlwind, at least, a much more complex relationship between funding justifications and technology obtained. Their studies of possible military applications and their contacts with military agencies expanded the Whirlwind group's sense of possibilities and unsolved technical problems. At the same time, they served to educate the funding agency about as yet undreamt-of possibilities for centralized command and control. While the ONR was not ultimately convinced, the thinking and the documents produced in this exchange kept funding going for several years and later proved of enormous value in convincing another military agency, the Air Force, to offer support. The source of funding, the political climate, and their personal experiences directed the attention of Forrester's group toward military applications, while the group's research eventually directed the military toward new concepts of command and control.

We could call this a process of *mutual orientation,* in which each partner oriented the other toward a new arena of concerns and solutions. Negotiations over funding, at least in this case, became simultaneously negotiations of the eventual technical characteristics of computers and of military command structures and strategic goals.

Through this process, within the space of a very few years the Air Force traditionalists who had opposed the computerized air defense system either became, or were replaced by, the most vigorous proponents of high-technology, computerized warfare anywhere in the American armed services.

Finally, SAGE set a pattern, repeated incessantly in subsequent years, of computerized command and control of nuclear defenses. Over two dozen large-scale, computerized, centralized command-control networks were built by the Air Force between the late 1950s and the middle 1960s, the so-called "Big L" systems, including the Strategic Air Command Control System and the Ballistic Missile Early Warning System (Bracken 1984). In 1962 the World-Wide Military Command and Control System, a global network of communications channels including (eventually) military satellites theoretically enabling central, real-time command of American forces worldwide, became operational.[4] The distant early warning systems used by

SAGE were ultimately connected with central computer facilities at the headquarters of the North American Air Defense Command in Colorado for ICBM detection and response.[5] President Reagan's Strategic Defense Initiative was thus only the latest in a long series of computer-controlled, centrally-commanded schemes for total defense (Edwards 1987, 1989; Edwards forthcoming; Franklin 1988). In this sense, SAGE technology had major impacts on military doctrine and organizational structure. SAGE technology was also used by IBM to build the Semi-Automatic Business-Research Environment (SABRE) — a direct reference to SAGE — the first computerized, centralized airline reservation system.

Computers and Work: Banking and the "Productivity Puzzle"

Computers have had equally massive effects on the nature, quality, and structure of work, where they are said to be largely responsible for the emergence of "post-industrial society" and for an "information revolution." Here, too, we find that an ideology of technological determinism is commonplace, reflected in managers' frequent belief that both productivity gains and social transformation will be automatic results of computerization. This section attempts to balance this view against the idea of a "web of computing," in which computers are only one of a variety of social and technical factors affecting organizational efficiency and culture (Kling 1982).

Figures 1 and 2 show the dramatic trends in expanding computing power and decreasing cost per computation (note the logarithmic scales of both charts). Computers began to be widely used in non-military industry and business in the late 1950s. At that point they were still so expensive that only large corporations could afford them. A decade later they were enough smaller and cheaper to be practical for middle-sized firms, and by the end of the 1970s almost any business with significant data processing needs either owned or leased a computer. In the past decade, of course, the introduction of personal computers (PCs), workstations, and powerful networking devices has put computers on the desks of a huge proportion of the American work force, especially in offices. Somewhere in the range of 50 million personal computers are installed in American homes, offices, and schools, as well as millions of other kinds of computers and terminals (Dertouzos 1991, p. 63).

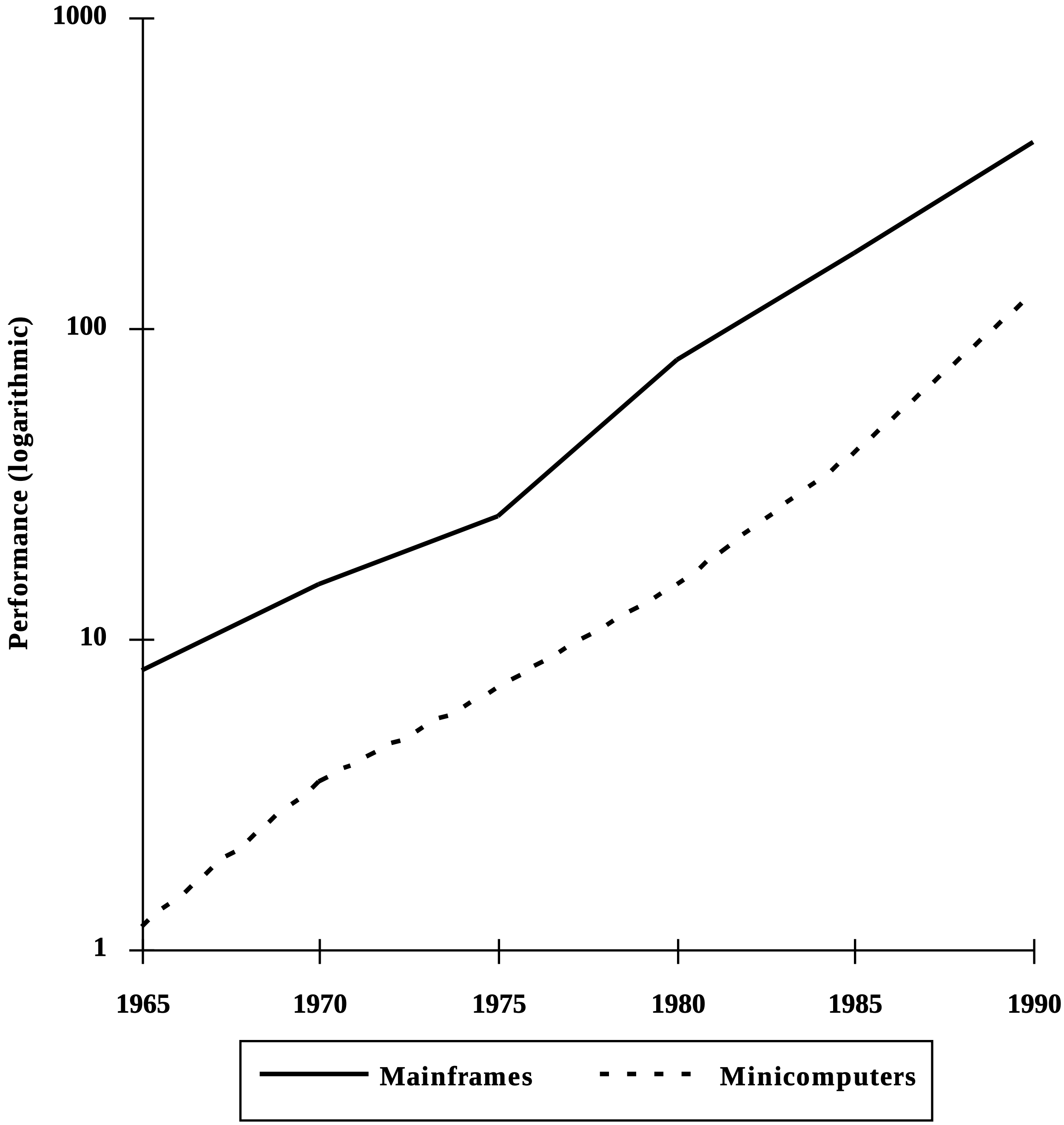

Figure 1. Computer Performance Growth 1965-1990.
Data: Jack Worlton, Los Alamos National Laboratories.

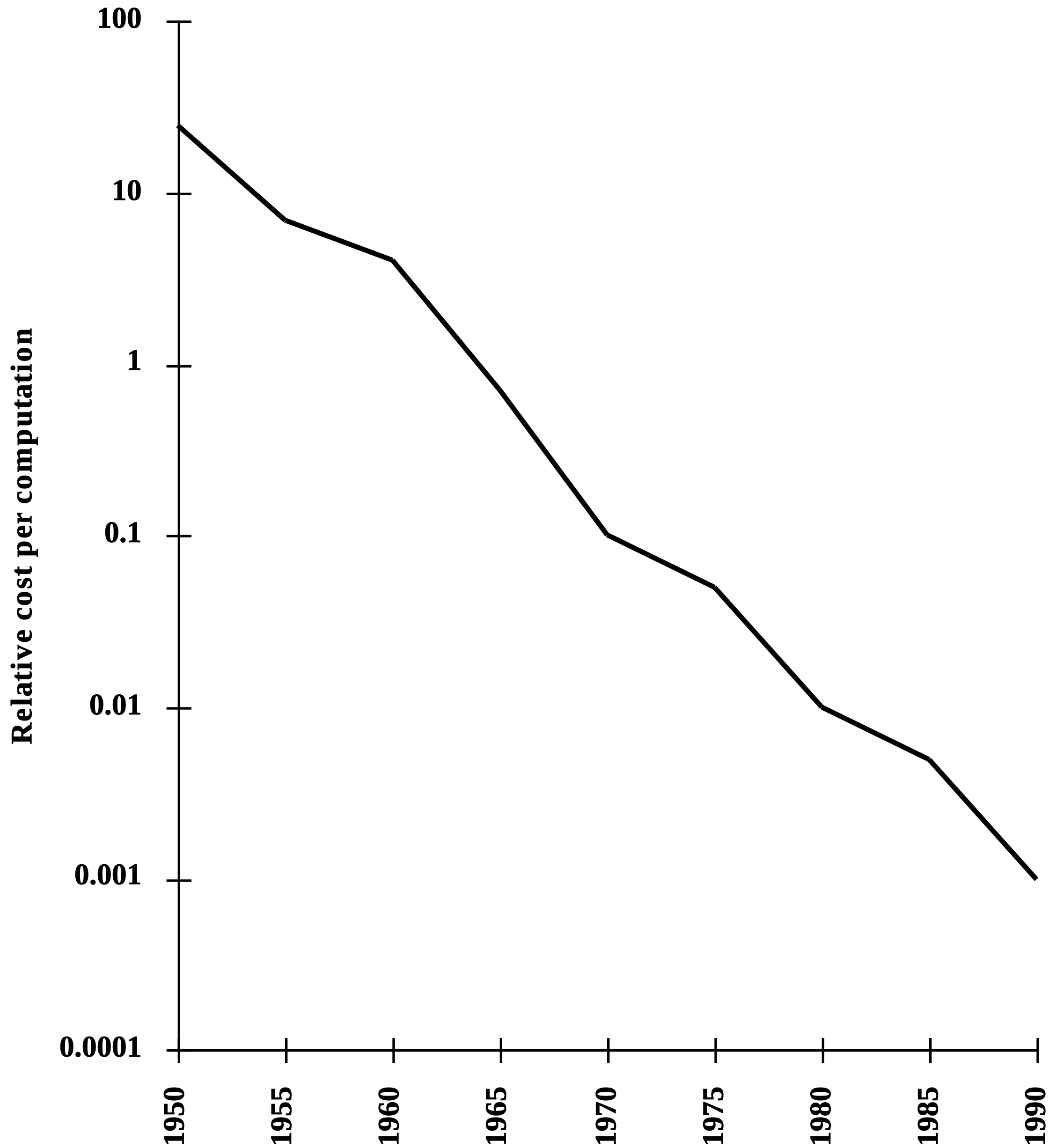

Figure 2. Declining cost of computation, 1950-1990. Based on cost per computation of most powerful commercial computers of each era. Source: Victor Petersen, NASA Ames. *Scientific American,* September 1991.

Along with computing and communications technologies have come dramatic increases in the size of the service and information sectors of the economy, as shown in Figure 3. It is often assumed, common-sensically, that the reason for the rush to computerize must lie in the benefits of computers to productivity (defined as the ratio of output to hours worked), and indeed automation is frequently urged as the key to productivity growth (Cohen and Zysman 1987). But despite an enormous scale of investment, the expected benefits have materialized in a way that must be characterized as *at best* spotty and fragile. Since the end of the 1960s American productivity growth has been weak, and with the rise of Japan in the late 1970s this became, and remains, a major policy concern (Baily 1991; Cohen and Zysman 1987).

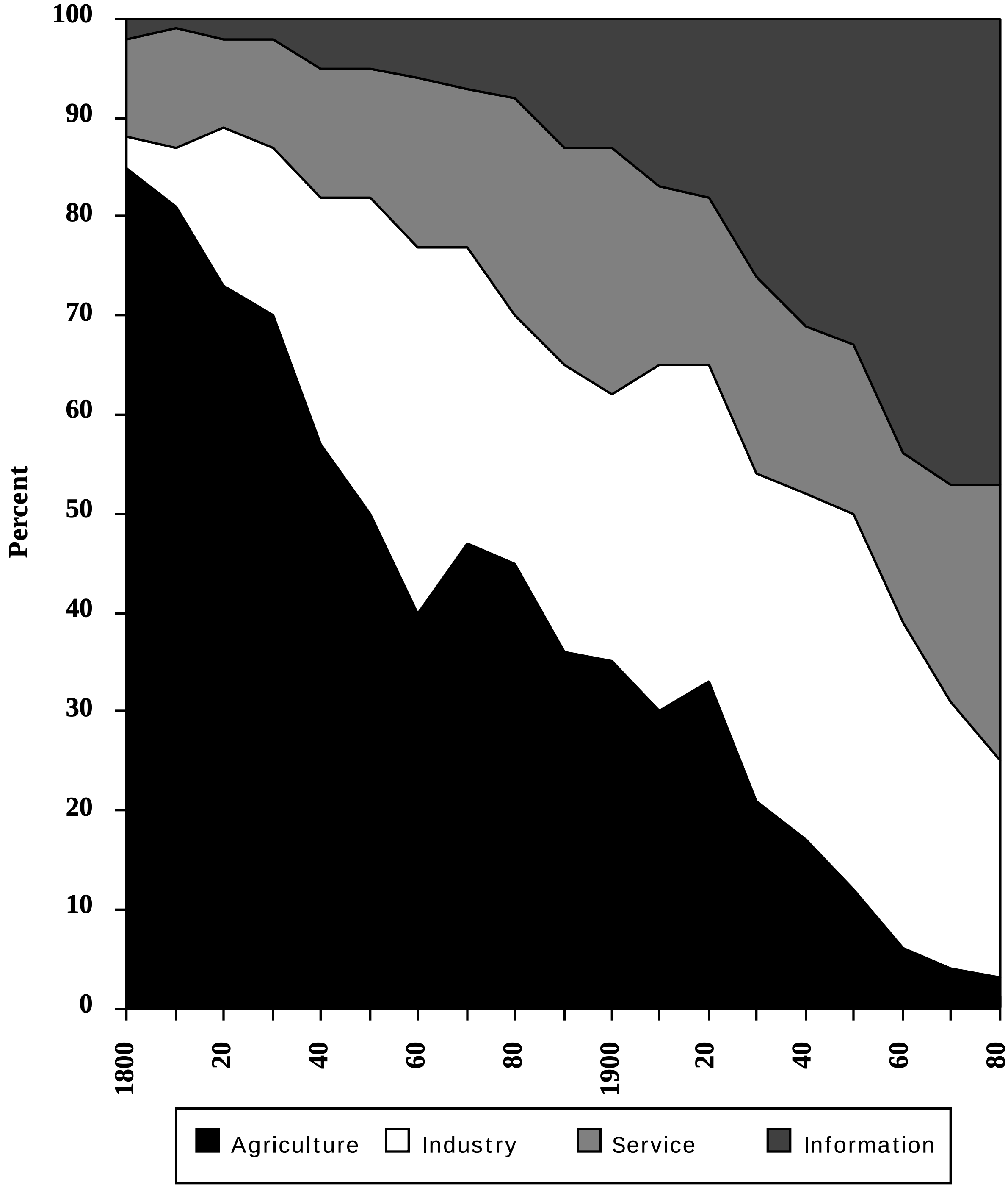

Figure 3. U.S. Civilian Labor Force by Sector, 1800-1980.
Source: Beniger (1986), p. 23.

The role of computers in this problem is a strange one. The computer manufacturing sector has been the *greatest* single contributor to productivity growth in American commerce. But in the heavily computerized service industries, productivity growth has been very poor. As Martin Baily observes, "Apparently we are getting better at making computers, but we still don't really know what to do with them once they're built" (Baily 1991, p. 112). This is what is known as the "productivity puzzle."

The example I will consider here is the banking industry, the first major non-military sector of the world economy to computerize. The entire business of banking is in effect a form of information processing, and traditional techniques made banking far more labor-intensive than the economy-wide average. It would appear, then, to be an ideal arena for computerization, one that could be expected to make fantastic gains from the automation of calculation, account management, billing, and check processing.

Interestingly, the first check-processing computer system, ERMA, was developed in a secret collaborative project between Bank of America and Stanford Research Institute. At the time of its public announcement in 1955 *no* other banks were investigating similar computerized systems. Yet histories of computers in banking frequently claim that computerization was "required" by rapidly increasing transaction volumes, labor costs, and high turnover of (primarily young, female) tellers and clerks (Fischer 1993; O'Brien 1968). ERMA introduced magnetic ink character recognition, which allowed partially automatic processing of checks. It initiated a huge wave of investment in computer equipment by the banking industry, one that continued such that 97 percent of commercial banks used computers by 1980. Richard Franke has studied the American financial industry to determine the effects of this investment on the industry's productivity and profitability (Franke 1989).

American banks: heavy investment, slow growth

Franke found that between 1948 and 1983, American banks' output rose fourfold, though the strongest period of output growth was 1948-58, *before* computers were introduced (see Figure 4). Labor input (that is, hours worked) also rose steadily, though more slowly, to three times its 1948 level. After 1958, labor input rose slightly *more* quickly, rather than less. And capital input — as might be expected — rose to 14 times its 1948 level, jumping from a 2.7 percent per year rate of growth to a 9.1 percent rate after 1958, the bulk of the jump attributable to computing and its indirect effects, such as the increased convenience of branch banking.

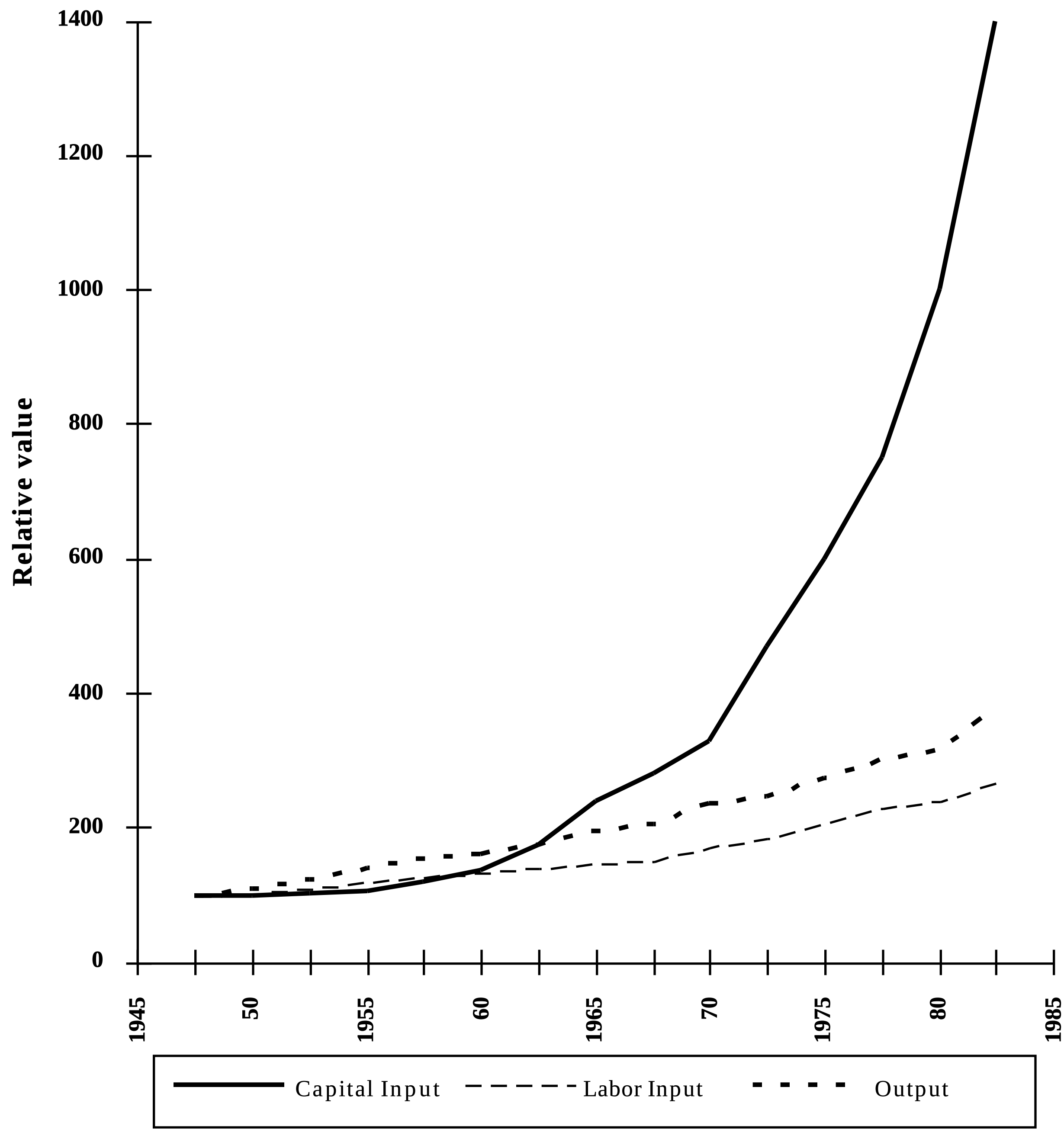

Figure 4. Financial industry inputs and output, 1948-1983.
Real terms, 1948 = 100. Source: Franke (1989), p. 284.

Yet this immense investment had virtually no effect on labor productivity. Figure 5 shows that productivity rose more quickly before 1958 than afterward, peaked in 1975, and declined slightly thereafter. This meant, of course, that while the capital intensity (the ratio of labor to capital inputs) of the industry quintupled, its *capital* productivity (the ratio of output to capital) declined to a mere one-fifth of its 1948 level. Data from the 1980s show productivity growing again — but only at the unimpressive rate of two percent per year.

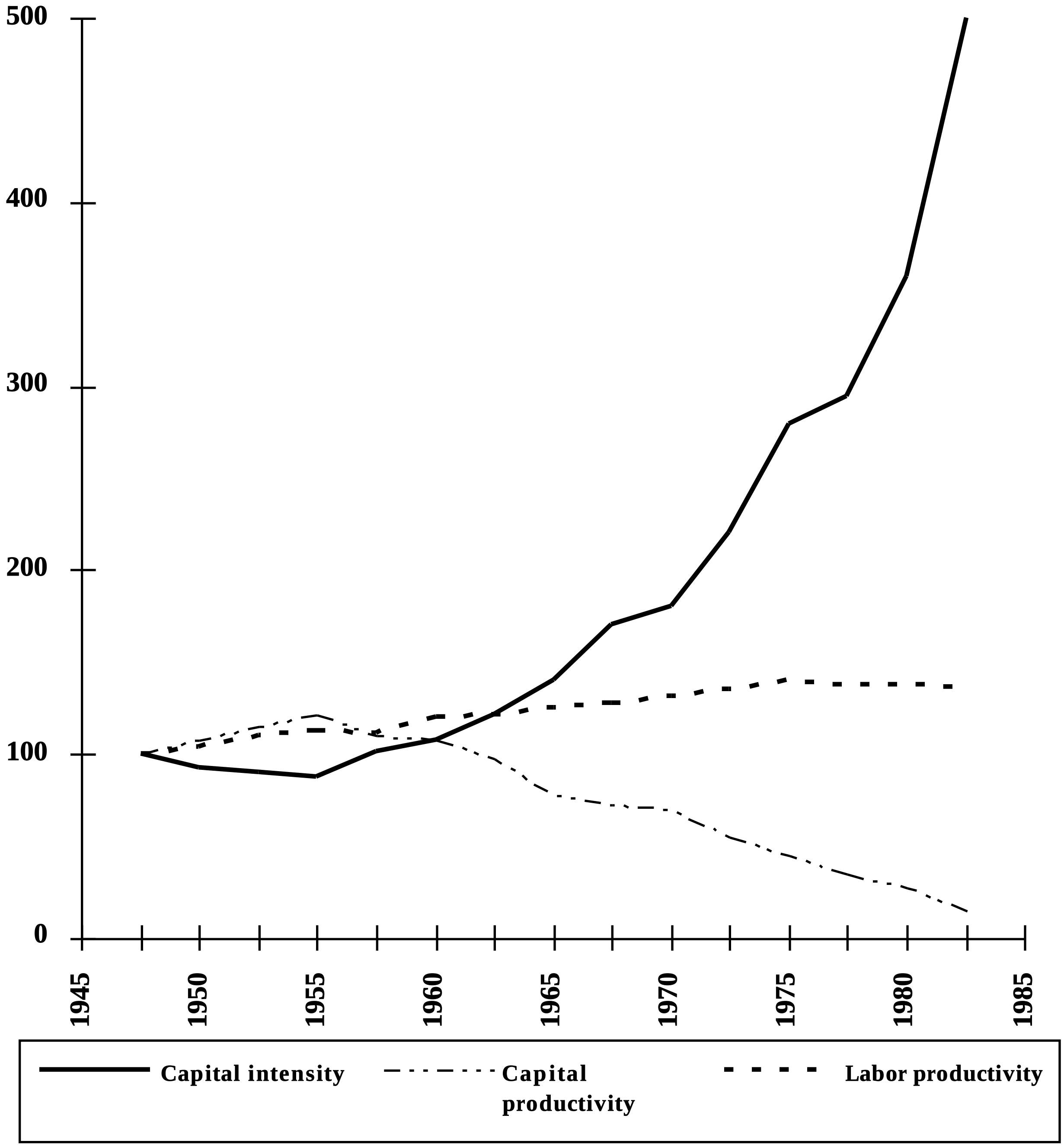

Figure 5. Capital and labor productivity in the financial industry, 1948-1983.
Real terms, 1948 = 100. Source: Franke (1989), p. 285.

This investment did, of course, take place during a period of very rapid technological change, when banks found themselves frequently replacing obsolete equipment that had been new a few years before. However, Franke used statistical regressions to allow for these effects and still found that productivity did not begin to improve until the fourth generation of computer technology, and even then not much.

What explains the paradox of massive automation with nearly nil results?

One possibility is Franke's own view. He concludes that "[f]undamental changes in the distribution and organization of work, *due to the new technology*, result initially in diseconomies. Only with time can enterprises adjust to become productive" (Franke 1989, p. 288, my italics). This macroeconomic explanation relies on the familiar "impact" model of the relations between technology and society: computers, colliding with the banking industry, split it apart like a fissioning atom which is only now beginning to restabilize into a new coherence. "Diseconomies" were the result.

But a look at the micro level — at what has actually happened in individual banks — shows that a diametrically opposite explanation may, at least in some cases, be more appropriate.

"Global Bank Brazil"

Shoshana Zuboff, who carried out detailed longitudinal studies of computerization in several factory and office settings between 1982 and 1986, examined the development of a data-base environment at the Brazilian branch of a major US bank (Zuboff 1988). She calls the institution "Global Bank Brazil."

At Global Bank Brazil, a group of far-sighted young managers had determined to leapfrog other banks by developing and installing an information system. The new computers would allow them not only to automate existing procedures, but to develop and sell a wide range of new information-based products. For example, they envisioned integrated real-estate sales in which the bank would provide a "package" of information about properties, loans, and insurance, or "smart" loan brokerage based on continually updated knowledge of clients' cash positions. The bank's computers would link one company's need for cash with another's excess, and bankers would mediate the deal.

These same managers also subscribed to an "impact" view of the database environment. They believed that once installed, bankers would automatically become more involved in analysis and decision-making based on information the system provided. Instead of spending their time on the phone or golfing with clients, maintaining personal relationships and getting an intangible “gut feeling” for their situations, bankers would work with hard data. The key to their new jobs would be effective exploitation of information.

In the words of one manager,

> Service, excellence, and innovation are only buzzwords right now. As we push the technology, people will realize that they have a really valuable tool on their hands. Then they’ll be forced to use it. Then we can change the way they think and do their work.

Another said,

> We’re on a learning curve now, trying to understand the technology. But at some point we’ll have a revolution. The technology will prove that the current organization is inadequate. Some people will accommodate to the new environment, and some won’t. In every revolution a lot of people are killed. And some people will be dead at the end of this one, too (Zuboff 1988, p. 214).

But instead of causing a “revolution,” the database environment became mired in an institutional backwater, automating some routine clerking tasks and having very little effect on the way the bank did business.

The reason had to do with the fact that senior managers, from the beginning, had been resistant to the data-base project. In order to avoid the senior managers’ interference and the watering down of their own “revolutionary” goals, the data-base developers had decided on an implementation strategy that would sneak the technology in through the bank’s figurative back door. Instead of installing it first in the bank’s marketing department or some other high-visibility area, they chose to introduce it into the central liabilities section, the oldest, least automated, and one of the most deeply internal of the bank’s operations — a “back office.”

Central liabilities maintained records of customers’ credit balances and client histories. Here the database served simply to automate an existing task. Clerks were trained to enter data on the new system, but were not told how it functioned. The algorithms it used were deemed too difficult for clerks to understand, and even the meaning of the term “database environment” was never explained to the group. A cursory training period totaling eight days left no one in the department in a position to understand the “revolutionary” potential on which the designers relied.

In consequence, the database became understood by those not privy to the designers’ goals as a control function, not a product development function.

Worse, it was associated with the dreariest of the bank's tasks. The project, still going on when Zuboff's study ended in 1984, had stalled far short of its original visionary goals.

The project managers had chosen this arena because they believed the technology would force a reorganization of the bank's functional divisions and power structures. But the implementation strategy they chose produced a particular *social* role for the new system. They isolated themselves from the bank's senior management, and effectively concealed the nature of their project even from its first users. Relying on an impact model of social change, the database developers avoided raising organizational issues directly — and guided their project into an organizational black hole. They did not understand that the database "environment" was not self-contained, but only one element of a larger socio-technical system that Kling and Scacchi have called the "web of computing" (Kling and Scacchi 1982).

Let us now look at an opposite case, where system developers understood very well the social purpose of what they were doing but failed to take into account some of the social impacts of the technology. This case is the computerization of British banking in the 1970s and early 1980s.

British banks: computers as strategies for organizational change

In British banking the traditional mode of training prior to computerization was based on a master-apprentice model, according to Steve Smith (1989). Employment began at age 15 or 16, and one then rose level by level through a pyramidal hierarchy. Ultimately, with luck and aptitude, any employee could hope to become manager of a branch bank or even a general manager at corporate headquarters. Branch banks under the old system were full-service banks under a decentralized corporate system. Branch managers, by virtue of their apprenticeships, were capable (at least in theory) of performing any operation at any level of the branch's hierarchy. Senior managers were thus generalists whose decision-making skills and authority were held to result from a broad and deep personal experience.

Along with this career structure went an ethos of employee flexibility. Clerks had a relatively wide range of skills, allowing them to shift from task to task during the banking day, which might require posting of transactions and billing in the morning, when few customers were coming in, and cashiering toward the end of the day, when customers came in to cash paychecks and withdraw funds.

Computerization, in this case, was introduced largely *in order* to restructure work. Smith quotes the managing director of Olivetti to the effect that

> [I]nformation technology is basically a technology of coordination and control of the labor force, the white-collar workers, which Taylorian [i.e. F. W. Taylor's scientific management methods] does not cover... [E]lectronic data processing (EDP) seems to be one of the most important tools with which company management institutes policies directly concerning the work process conditioned by complex economic and social factors. In this sense EDP is in fact an organizational technology, and like the organization of labor, has a dual function as a productive force and a control tool for capital (Franco de Benedetti, 1979, cited in Smith 1989, p. 383).

British bankers installed computers as part of a general plan to move away from the craft-apprenticeship model toward a rationalized industrial-production model. Computers facilitated, for example, progressive specialization of tasks and automation of a great deal of work once done by hand.

Along with this specialization went a deliberate restructuring of career paths. Today not one but several tiers of entry are recognized, and more horizontal and vertical segmentation of functions has occurred, resulting in a more differentiated structure where not all paths begin at the bottom or lead to the top, and more specialized jobs mean greater expertise but less flexibility.

Some banks also used computers to centralize operations into a hub-and-satellite configuration called “branch network reorganization.” Satellite branches, in the new scheme, offer limited services, mostly to individuals. Some satellites have no managers. The central office houses the data processing services as well as specialized services for corporate clients and investors. This centralization reinforces the segmentation of banking work and creates a class of specialist managers.

But the outcome of this computer-based restructuring of bank organization was mixed. While productivity in such repetitive tasks as data entry rose, numbers of clerical staff did not decline and frequently rose. A new gender division of labor also emerged, with more women working in the low-ceiling role of clerks and men clustering in what was known as the “accelerated career program.” Smith cites the frequent “complaint that staff who joined to be bank employees find themselves ‘dedicated’ to repetitive ‘factory work’: ‘This isn’t banking, it’s factory work’” (Smith 1989, p. 385).

The repetitive nature of more segmented work, together with the corresponding reduction in sense of collectivity and community, caused declines in morale in some (not all) banks. This finding has been replicated in other studies of computerization in office work where managers’ goals have been similar (Attewell 1987; Garson 1988; Zuboff 1988). The decreased flexibility of less-skilled workers led to inefficiencies because of the variable work structure of the banking day. Finally, tensions arose between old-school generalist managers and the younger specialists over the very nature of banking. The younger group tended to treat branch operations as mechanical or industrial processes. Generalists felt this as an insult to a formerly dignified career and also believed that the younger group lacked an intuitive understanding of bank operations, relying too heavily on analysis. The overall result, as in the American case, was a surprisingly low growth in productivity.

The British case shows computers used to facilitate the creation of a social product, in this case an automation of traditional craft work and a centralization of the formerly decentralized branch system. These ends were embodied in the design of the computing systems they incorporated, especially in the centralization of data processing (reducing branch banks to input-output devices) and the segmentation of work, with data entry tasks separated from other, more complex banking assignments. As the reorganization and the investment in computing equipment proceeded, they had impacts upon the social space of the organization — many of which were neither foreseen nor desired by the designers. “Rationalizing” an existing process introduced new irrationalities, partly because it treated the organization as a machine without taking into account such social factors as job satisfaction and gender, and partly because computer systems rigidified a less flexible work structure (Kling and Iacono 1984; Kling and Scacchi 1982).

From these two examples we can see that there may be sociocultural as well as technological-economic reasons for the productivity puzzle. When computers were introduced at Global Bank Brazil in hopes they would "impact" the organization, the result was partial failure due to inertia — a failure to treat the social context directly. When computers were introduced as part of a direct treatment of the organizational context, based on an automation model, they had unforeseen impacts on the culture of work that led to inefficiencies and social dislocations.

Gender and Computers

Computer work is stratified in an almost linear way along an axis defined by gender. Women are overwhelmingly dominant in the lowest-skill, lowest-status, and lowest-paid areas, such as microchip manufacture and computer assembly (especially in "offshore," or foreign, factories) and data entry, where women account for up to 95 percent of the workforce. While statistical evidence in this area is problematic, a general trend is unmistakable: numbers of women begin to decline as skill levels rise, with somewhere on the order of 65 percent of American computer operators, 30-40 percent of programmers, and 25-30 percent of systems analysts being female. Gender imbalances in European countries are more dramatic (Frenkel 1990; Gerver 1985).

A similar pattern exists in education, in a way that closely parallels gender differentiation in mathematics. Girls and boys display roughly equal interest and skill in the primary grades, but starting around age 11 or 12 girls begin gradually to stop enrolling in computer courses. By high school age boys outnumber girls in such courses roughly two to one. During the 1980s roughly this same ratio of men to women persisted through undergraduate college, with about 35 percent of bachelor's degrees in computer science awarded to women. But there is some evidence that this ratio has declined substantially, perhaps to as little as 20 percent, in the last two or three years, without a corresponding drop in other technical majors.[6]

By the Ph.D. level the situation is much more dramatic: the percentage of computer science Ph.D.'s awarded to women has remained steady at 10-12 percent since 1978. The situation in engineering is worse, with women receiving only 8 percent of Ph.D.'s, though the numbers there have been rising. For comparison, note that the percentage in the physical sciences and mathematics is now about 17 percent and rising.

The imbalance is most severe at the level of faculty employment. Only 6.5 percent of tenure-track faculty in computer science departments are female (7 percent in computer science and 3 percent in electrical engineering). One-third of Ph.D.-granting departments have no women faculty at all.

Sexism in educational settings

One possible version of this story relies for an explanation on bias and systematic oppression. High-school age boys have frequently been observed to harass girls and demean their skills, sometimes deliberately in order to keep enrollments in computer classes low. Illustrations in computer science textbooks typically show a ten-to-one ratio of men to women, and computer advertising is strongly male-oriented. Women students at all levels have reported oppression in many forms, ranging from overt statements by senior professors that women do not belong in graduate school to more subtle and probably unconscious mistreatment, such as seeing their own ideas ignored or patronized in the classroom while similar ideas of their male colleagues receive praise. The following quotations from students and research staff illustrate the sometimes very direct nature of this sexism.

> While I was teaching a recitation section, a male graduate student burst in and asked for my telephone number. Men often interrupt me during technical discussions to ask personal questions or make inappropriate remarks about nonprofessional matters.
>
> I was told by a secretary planning a summer, technical meeting at an estate owned by MIT that the host of the meeting would prefer that female attendees wear two-piece bathing suits for swimming.
>
> I was told by a male faculty member that women do not make good engineers because of early childhood experience... little boys build things, little girls play with dolls, boys develop a strong competitive instinct, while girls nurture... (anonymous interviewees, cited in Frenkel 1990, pp. 36-7).

Such factors as the lack of female role models and the so-called “impostor” phenomenon, in which minorities feel themselves not to be “real” members of the dominant group, distrusting their own skills and avoiding public display so as not to be caught out “impersonating” a “real” computer scientist, are among the other ways gender stratification perpetuates itself (Leveson 1989; Pearl and others 1990; Weinberg 1990).

These are real and important mechanisms in creating gender imbalance. At the same time, there is evidence to suggest that in the computer industry, far from a systematic exclusion, many companies have made active efforts to recruit more women, and that compared with other, older industries, computing has been a more favorable environment for women. In academia, the very scarcity of women Ph.D.'s makes finding qualified candidates difficult. So while more subtle bias persists, *direct* discrimination against women is probably somewhat less of a factor in computing than in other careers (Leveson 1989).

Cultural construction and gendered tools

But another approach to the issue of gender differences is to ask the question of whether or not computers, *as tools*, are gender-neutral. I will argue that they are not: in fact, computers are culturally constructed in such a way as to stamp them with a gender and make them resistant to the efforts of women to "make friends" with them (Edwards 1990; Edwards forthcoming; Perry and Greber 1990; Sanders and Stone 1986).

Scientists tend to think of computers abstractly as Turing machines, universal machines capable of doing anything from controlling a spaceship to balancing a checkbook. But people always encounter technology in a particular context and develop their understanding from there. If they first meet computers in a course, they are likely to be introduced to them in a theoretical mode that emphasizes their abstract properties and their electronic functioning. If they meet computers in an office they may understand them as word processors or spreadsheet calculators. In every context they will be surrounded by a sort of envelope of other people's talk, writing, attitudes, images, and feelings about them. The formal content of a course or a training session or a conversation with another user is only part of what is communicated.

Many investigators have suggested that computer avoidance in girls is connected with differences between what can be loosely termed the "cultures" of men and women. (Of course there is great variability within the generalizations I am about to describe.) Men learn to value independence — the ability to do things on their own, without help. They are most comfortable in a social hierarchy in which their position is relatively clear. They are trained early on for roles as competitors and combatants, and they value victory and power. Abstract reasoning is, for men, an important value, partly because of its connection with power. Carol Gilligan's well-known study of men's and women's morality, *In a Different Voice*, revealed that men tend to see the highest form of morality as one based on a reasoned adherence to an overarching moral law that treats all actors as equals (Gilligan 1982).

Women, by contrast, tend to prefer interdependence. Reliance on others is valued because it continually maintains a social fabric or network, seen as more important than individual self-sufficiency. Instead of hierarchy, women’s culture practices social “leveling,” in which an underlying goal of conversations or games is to keep everyone at the same level of status. Similarly, competition and winning are less important than keeping a game or conversation going (Tannen 1990). Practical skills rather than abstract reasoning tend to be primary values, and this goes along with a morality that perceives particular relationships as superseding abstract rules — people are treated differently depending on their needs and relationships to other actors, rather than similarly based on their moral equivalence (Longino1990).

In her studies of children learning to program in the LOGO language at a private school, Sherry Turkle observed two basic approaches to computer programming. Students she calls "hard masters" employed a planned, structured, technical style, while "soft masters" relied on a more amorphous system of gradual evolution, interactive play, and intuitive leap. In her words,

> hard mastery is the imposition of will over the machine through the implementation of a plan. A program is the instrument for premeditated control. Getting the program to work is more like getting ‘to say one's piece’ than allowing ideas to emerge in the give-and-take of conversation. ...[T]he goal is always getting the program to realize the plan.
>
> Soft mastery is more interactive... the mastery of the artist: try this, wait for a response, try something else, let the overall shape emerge from an interaction with the medium. It is more like a conversation than a monologue (Turkle 1984, pp. 104-5).

Note the similarity of these two modes with the two cultures I have described. In fact, Turkle found, the majority of hard masters were boys, and the majority of soft masters were girls. But both styles produced some consummate programmers.

Both Turkle’s hard and soft mastery and my descriptions of men’s and women’s cultures are, of course, caricatures of immensely flexible and complicated processes rather than hard-and-fast rules. A culture is not a program, but a subtle set of nudges in particular directions which not everyone receives to the same degree or responds to in the same way. Many men are more at home in what I have described as “women’s” culture, and vice versa. Some learn to be equally at home in both modes. And it is important that excellent programs can be written by people of both sexes using both methods, something Turkle saw in men and women of all ages (Turkle 1984; Turkle and Papert 1990).

Nevertheless, these two dichotomies are suggestive.

Consider, for example, the fact that many if not most video games emphasize violence, often with a military metaphor. The first video game was "Space War," written by MIT hackers during the early 1960s (Levy 1984). (But the first commercial game was the benign "Pong," and one of today's most popular games is the equally unmilitaristic Tetris.) Still, the great bulk of the games that led the video arcade craze of the early 1980s were combative in nature, and it was partly as a belated response to the potential market among adolescent girls that less-violent alternatives such as Frogger and Pac-Man were introduced.

Hacker culture, to give another example, is strongly male-oriented. Hackers frequently work in independent isolation. Many say their fascination with hacking is related to the sense of control and power, an elation in their ability to make the machine do anything (Weizenbaum 1976; Hafner 1991). While the so-called "hacker ethic" described by Steven Levy theoretically values programming skill above all else including physical appearance and gender, in practice hackers frequently avoid women and exclude them from their social circles (Levy 1984; Turkle 1984). Turkle's ethnographic study of MIT hackers revealed a powerful competitive side in such phenomena as "sport death," the practice of staying at one's terminal until one drops, achieving fame through a kind of monumental physical self-denial. In the 1960s and 1970s, and to some extent still today, hackers played an important unofficial role in the development of system software and computer games. So their conceptions of the nature of computing were, in a sense, embodied in machines.

Another source of gender differentiation may be the nature of computer instruction in schools and colleges. Computer science, with its marginal disciplinary position between mathematics, cognitive psychology, and engineering, has to a certain extent relied for institutional survival on laying a claim to mathematical-scientific purity, and one place this claim is expressed (and students are weeded for correct skills and orientations) is introductory computer science courses. Traditional programming courses, partly for this reason, are taught in a highly theoretical mode which emphasizes abstract properties of logic, computation, and electronics rather than practical uses. Girls report disinterest and frustration in classes with this orientation and get better grades in courses with a more practical bent.

In a major 1989 debate in the pages of the main computer science journal, Communications of the ACM, University of Texas at Austin computer scientists Edsger Dijkstra proposed that introductory computer science be taught in an even more formal model, emphasizing its fundamentally mathematical core (Dijkstra 1989). Rather than use real computers, students in Dijkstra's program would have to write programs in unimplemented languages and prove their validity logically (instead of debugging them by trial and error methods). Many of his colleagues objected to this excessively formalistic view — but it unquestionably reflects one important strand of thought about computer learning. To the extent that this teaching strategy holds sway, it tends to inhibit women's entry into the field (Frenkel 1990).

These last three examples — video games, hacking, and computer instruction — all show the process of cultural construction in action. Interaction with combative video games constructs the computer as a site of conflict and competition, a game where winning is a matter of metaphorical life and death. Hacking uses the computer as a medium for a social process of self-construction in which young men compete with each other and with the machine and achieve independence and power. The computer, as the site of this self-construction, receives a gender association. Computer instruction that emphasizes abstract rationality is more appealing for boys and facilitates the association of computers with men. Thus computers have frequently been culturally constructed as male-gendered objects.

Here, then, is another social process through which computer technology and the social production of knowledge and values interact. The tendency is to think of these cultural factors, because they are so flexible and variable, as separate from and independent of design. But people encounter them in their experience of computing as necessary presences which structure the computer they perceive. Social "context" and design interpenetrate; no element is purely essential and no others purely accidental (Winograd 1987).

Conclusion

These three brief case studies bring into relief the interaction of technology with politics, society, and culture as computers increasingly permeate industrialized societies. Computers rarely "cause" social change in the direct sense implied by the "impact" model, but they often create pressures and possibilities to which social systems respond. Computers affect society through an interactive process of social construction.

Who will get computers? What new kinds of access to information will they allow? Who will benefit, and whose activities will be subject to more detailed scrutiny? How will these actors react to such changes? These questions are especially important precisely because the computer is not only inserted into an organization or a culture, but frequently embodies particular images of how the organization or culture functions and what the roles of its members should be. Once introduced, a computer system, by embodying these images them, can help institutionalize and rigidify them. What is needed an awareness of the "web of computing" (Kling and Scacchi 1982), that is, of the ways in which a new computer system will be inserted into an existing network of social relationships. Neither a "social impacts" nor a "social products" approach will produce an adequate picture of this interaction; only an image of technological change as a social process is likely to be robust enough to capture the flavor of how computers work in society.

References

Attewell, Paul. “Big Brother and the Sweatshop: Computer Surveillance in the Automated Office.” Sociological Theory 5 (1987): 87-99.

Baily, Martin. “Great Expectations: PCs and Productivity.” In Computerization and Controversy, ed. Charles Dunlop and Rob Kling. 111-117. New York: Academic Press, 1991.

Beniger, James R. The Control Revolution: Technological and Economic Origins of the Information Society. Cambridge, MA: Harvard University Press, 1986.

Berkeley, Edmund C. The Computer Revolution. Garden City, NY: Doubleday, 1962.

Bijker, Wiebe, Thomas P. Hughes, and Trevor Pinch, ed. The Social Construction of Technological Systems. Cambridge, MA: MIT Press, 1987.

Brown, George W., interviewed by Richard R. Mertz 3/15/73. Smithsonian Computer Oral History, AC NMAH #196. Archive Center, National Museum of American History, Washington, D.C.

Brueckner, Leslie and Michael Borrus. “The Commercial Impacts of Military Spending: The VHSIC Case.” In Strategic Computing: Defense Research and High Technology, ed. Paul N. Edwards and Richard Gordon, unpublished ms.

Bush, Vannevar. Science: The Endless Frontier. Washington, D.C.: US Government Printing Office, 1945.

Callon, Michel. “Society in the Making: The Study of Technology as a Tool for Sociological Analysis.” In The Social Construction of Technological Systems, ed. Wiebe E. Bijker, Thomas P. Hughes, and Trevor Pinch. Cambridge, MA: MIT Press, 1987.

Cohen, Stephen S. and John Zysman. Manufacturing Matters: The Myth of the Post-Industrial Economy. New York: Basic Books, 1987.

Dertouzos, Michael L. and others. “Communications, Computers, and Networks: Special Issue.” Scientific American 265 (3 1991): 62-164.

Dickson, David. The New Politics of Science. New York: Pantheon, 1984.

Dijkstra, Edsger. “On the Cruelty of Really Teaching Computing Science” (with replies by others). Communications of the ACM 32 (12 1989): 1398-1414.

Edwards, Paul N. “A History of Computers in Weapons Systems.” In Computers in Battle, ed. David Bellin and Gary Chapman. 45-60. New York: Harcourt, 1987.

Edwards, Paul N. “The Closed World: Systems Discourse, Military Policy, and Post-WWII US Historical Consciousness.” In Cyborg Worlds: The Military Information Society, ed. Les Levidow. 135-158. London: Free Association Books, 1989.

Edwards, Paul N. “The Army and the Microworld: Computers and the Militarized Politics of Gender.” Signs 16 (1 1990): 102-127.

Edwards, Paul N. The Closed World: Computers and the Politics of Discourse. Cambridge, MA: MIT Press, forthcoming.

Fagen, M. D. A History of Engineering and Science in the Bell System. Murray Hill, NJ: Bell Telephone Laboratories, 1978.

Fischer, A. W. and J. L. McKenney. “The Development of the ERMA Banking System: Lessons from History.” IEEE Annals of the History of Computing 15 (1 1993): 44-57.

Flamm, Kenneth. Targeting the Computer: Government Support and International Competition. Washington, D.C.: Brookings Institution, 1987.

Flamm, Kenneth. Creating the Computer. Washington, D.C.: Brookings Institution, 1988.

Forman, Paul. “Behind quantum electronics: National security as basis for physical research in the United States, 1940-1960.” Historical Studies in the Physical and Biological Sciences 18 (1 1987): 150-183.

Forrester, Jay W., Hugh R. Boyd, Robert E. Everett, and Harris Fahnestock. A Plan for Digital Information Handling Equipment in the Military. MIT Servomechanisms Laboratory, 1948. Project DIC 6345.

Forrester, Jay W. Computation Books 1946-48. MIT Archives, Cambridge MA.

Franke, Richard H. "Technological Revolution and Productivity Decline: The Case of US Banks." In Computers in the Human Context, ed. Tom Forester. 281-290. Cambridge, MA: MIT Press, 1989.

Franklin, H. Bruce. War Stars: The Superweapon and the American Imagination. New York: Oxford University Press, 1988.

Frenkel, Karen A. "Women & Computing." Communications of the ACM 33 (11 1990): 34-46.

Garson, Barbara. The Electronic Sweatshop. New York: Simon and Schuster, 1988.

Gerver, Elisabeth. Humanizing Technology. New York: Plenum Press, 1985.

Gilligan, Carol. In a Different Voice. Cambridge, MA: Harvard University Press, 1982.

Goldstine, Herman. The Computer from Pascal to von Neumann. Princeton, NJ: Princeton University Press, 1972.

Gray, Chris. "The Cyborg Soldier: The US Military and the Post-Modern Warrior." In Cyborg Worlds: The Military Information Society, ed. Les Levidow and Kevin Robins. London: Free Association Books, 1989.

Gray, Chris Hables. "Computers as Weapons and Metaphors: The US Military 1940-1990 and Postmodern War." Ph. D., University of California at Santa Cruz, 1991.

Gray, Chris Hables. "The Cyborg Soldier: The US Military and the Post-Modern Warrior." In Cyborg Worlds: The Military Information Society, ed. Les Levidow and Kevin Robins. London: Free Association Books, 1989.

Gray, Chris Hables. "Artificial Intelligence and Real Postmodern War." unpublished Ph.D. qualifying essay, University of California, Santa Cruz, 1988.

Hafner, Katie and John Markoff. Cyberpunk: Outlaws and Hackers on the Computer Frontier. New York: Simon and Schuster, 1991.

Herken, Gregg. Counsels of War. New York: Knopf, 1983.

Hodges, Andrew. Alan Turing: The Enigma. New York: Simon and Schuster, 1983.

Hughes, Thomas P. "The Evolution of Large Technological Systems." In The Social Construction of Technological Systems, ed. Wiebe E. Bijker, Thomas P. Hughes, and Trevor Pinch. Cambridge MA: MIT Press, 1987.

Jacky, Jonathan. "Software Engineers and Hackers: Programming and Military Computing." In Strategic Computing: Defense Research and High Technology, ed. Paul N. Edwards and Richard Gordon. Unpublished ms.

Jacobs, John F. "SAGE Overview." Annals of the History of Computing 5 (4 1983): 323-329.

Kling, Rob and Suzanne Iacono. "Computing as an Occasion for Social Control." Journal of Social Issues 40 (3 1984): 77-96.

Kling, Rob and Walt Scacchi. "The Web of Computing: Computing Technology as Social Organization." Advances in Computers 21 (1982): 3-85.

Knorr-Cetina, Karin D. "Introduction: The micro-sociological challenge of macro-sociology: towards a reconstruction of social theory and methodology." In Advances in social theory and methodology: Towards an integration of micro- and macro-sociologies, ed. Karin D. Knorr-Cetina and Aaron V. Cicourel. 2-47. Boston: Routledge and Kegan Paul, 1981.

Latour, Bruno. Science in Action: How to follow scientists and engineers through society. Cambridge, MA: Harvard University Press, 1987.

Leveson, Nancy. Women in Computer Science. National Science Foundation, 1989.

Levy, Steven. Hackers. New York: Anchor Press, 1984.

Longino, Helen. Science as Social Knowledge : Values and Objectivity in Scientific Inquiry. Princeton, NJ: Princeton University Press, 1990.

Mahoney, Michael S. "The History of Computing in the History of Technology." Annals of the History of Computing 10 (2 1988): 113-125.

National Science Foundation. Women and Minorities in Science and Engineering.1990. NSF 90-301.

Noble, David. Forces of Production. New York: Oxford University Press, 1984.

O'Brien, James A. The Impact of Computers on Banking. Boston: Bankers Publishing Co., 1968.

Pearl, Amy, Martha E. Pollack, Eve Riskin, Becky Thomas, Elizabeth Wolf, and Alice Wu. “Becoming a Computer Scientist: A Report by the ACM Committee on the Status of Women in Computing Science.” Communications of the ACM 33 (11 1990): 48-57.

Perry, Ruth and Lisa Greber. “Women and Computers: An Introduction.” Signs 16 (1 1990): 74-101.

Pinch, Trevor and Wiebe Bijker. “The Social Construction of Facts and Artifacts, or How the Sociology of Science and the Sociology of Technology Might Benefit Each Other.” In The Social Construction of Technological Systems, ed. Wiebe Bijker, Thomas P. Hughes, and Trevor Pinch. 17-50. Cambridge, MA: MIT Press, 1987.

Redmond, Kent C. and Thomas M. Smith. Project Whirlwind: The History of a Pioneer Computer. Boston: Digital Press, 1980.

Rees, Mina. “The Computing Program of the Office of Naval Research, 1946-1953.” Annals of the History of Computing 4 (2 1982): 103-113.

Rosenberg, Nathan. “Civilian Spillovers from Military R&D Spending: The American Experience Since World War II.” Manuscript. Presented to the Conference on Technical Cooperation and International Competitiveness, Lucca, Italy, 1986.

Roszak, Theodore. The Cult of Information. New York: Pantheon, 1986.

Sanders, Jo Schuchat and Antonia Stone. The Neuter Computer: Computers for Girls and Boys. New York: Neal-Schuman Publishers, 1986.

Schaffel, Kenneth. “The US Air Force's Philosophy of Strategic Defense: A Historical Overview.” In Strategic Air Defense, ed. Stephen J. Cimbala. 3-22. Wilmington, DE: Scholarly Resources Inc., 1989.

Smith, Bruce L. R. American Science Policy Since World War II. Washington, D.C.: Brookings Institution, 1991.

Smith, Merritt Roe. Military Enterprise and Technological Change. Cambridge, MA: MIT Press, 1985.

Smith, Steve. “Information Technology in Banks: Taylorization or Human-centered Systems?” In Computers in the Human Context, ed. Tom Forester. 377-390. Cambridge, MA: MIT Press, 1989.

Tannen, Deborah. You Just Don't Understand: Women and Men in Conversation. New York: Morrow, 1990.

Turing, Alan. “On Computable Numbers.” Proc. London Math. Soc. 42 (2 1936): 230-265.

Turkle, Sherry. The Second Self: Computers and the Human Spirit. New York: Simon and Schuster, 1984.

Turkle, Sherry and Seymour Papert. “Epistemological Pluralism: Styles and Voices Within the Computer Culture.” Signs 16 (1 1990): 128-157.

Valley, George E., Jr. “How the SAGE Development Began.” Annals of the History of Computing 7 (3 1985): 196-226.

Weinberg, Sandy. “Expanding Access to Technology: Computer Equity for Women.” In Technology and the Future, ed. Albert H. Teich. 277-287. New York: St. Martin's Press, 1990.

Weizenbaum, Joseph. Computer Power and Human Reason: From Judgment to Calculation. San Francisco: W. H. Freeman, 1976.

Winograd, Terry. “Strategic Computing Research and the Universities.” In Computerization and Controversy, ed. Charles Dunlop and Rob Kling. 704-716. New York: Academic Press, 1991.

Winograd, Terry and Fernando Flores. Understanding Computers and Cognition. Reading, MA: Addison-Wesley Publishing Company, 1987.

Zuboff, Shoshana. In the Age of the Smart Machine: The Future of Work and Power. New York: Basic Books, 1988.

Notes

[1] The introduction to Beniger (1986) provides an interesting synoptic view of this literature, and the introduction to Dunlop and Kling (1991) gives a very helpful critical analysis. As Dunlop and Kling argue, notions of utopian and "revolutionary" effects — or their converse, the Orwellian idea of computerized Stalinism — have been substantially, even hysterically oversold. This is especially true in the areas of office automation and computing in government, where their effects on productivity and panoptic power have been considerably less than many imagine.

[2] Analog computation represents variables using continuous physical quantities such as electrical resistance, motor speed, or voltage, which are physically combined to yield a result. Everyday examples of analog devices are volume controls (variable resistors) and ordinary clocks (motor speed). The once ubiquitous slide rule is a commonplace example of an analog computer: mathematical operations are performed on numerical quantities (represented as positions along the length of the rule's scales) by sliding the rule's moving middle section back and forth. The rule's length is a continuous quantity. Digital computation represents variables as discrete quantities such as whole numbers, switch positions, or magnetic polarity. Everyday examples of digital devices are light switches (on or off) and digital clocks (which unlike ordinary clocks show hours and minutes as discrete, unit quantities).

[3] For fuller accounts of wartime and postwar developments, see especially Edwards (1987, 1989), Goldstine (1972), Redmond and Smith (1980), and Rees (1982). My account in the rest of this section relies heavily on Flamm (1987, 1988), who gives the best-informed history of military involvement, though his perspective is generally technical and economic. For a book-length social and cultural analysis, see Edwards (forthcoming).

[4] But see Jacky (unpublished ms.) for a description of the system's shaky record of reliability.

[5] Borning (1987) presents a lengthy history of NORAD computer failures, some serious enough to lead to escalations in the alert status of nuclear forces. Problems of complexity and reliability in these systems became a social trope for nuclear fear, as reflected in films and novels from *Dr. Strangelove*, *Fail Safe*, and *Colossus: The Forbin Project* in the 1960s to *War Games* and *The Terminator* in the 1980s, all of which involved some variation on the theme of computer-initiated nuclear holocaust.

[6] Most of the statistical information in this section is drawn from National Science Foundation (1990), Frenkel (1990), Gerver (1985), and Pearl et al. (1990).