Day 17: Smith & Clancey

Wednesday, November 4, 2020 3:39 PM

SmithClancey_Pg444...

normally performed by engineers just five or ten years ago. Engineers, in turn, are freed to delve into even more technically sophisticated areas. As a peripheral advantage, the critical need for technical manpower is partially satisfied.

To manage technological change, we must manage our human resources better. For example, we must commit ourselves to ensuring that none of our workers is laid off because of technological changes, as long as they are willing to be retrained and accept new job assignments. Our experience at Westinghouse has been that employees displaced by robots normally move up to better, more challenging work. . . .

In Westinghouse, by putting the programs for people first, we expect to multiply the productivity improvements that are gained through technology and capital investments. With participative management, for instance, employees welcome advanced technology because they feel in charge of it.

"The Race to Build a Supercomputer"

WILLIAM D. MARBACH

One day in 1981, Michael L. Dertouzos, the director of the renowned Computer Laboratory at the Massachusetts Institute of Technology, received from a colleague just back from Japan a draft of a research paper. It outlined Japanese proposals for long-range research projects in advanced computer science—plans to build revolutionary artificial intelligence computers and supercomputers a thousand times faster than today's machines. To Dertouzos's ears, the document had a familiar ring. "I looked at it and I started panicking, panicking, panicking," he recalls. "I said, 'My God, this is the research charter for my laboratory. These guys have stolen it.'"

The Japanese had in fact taken nothing but the initiative, yet Dertouzos immediately saw the threat: Japan's JIPDEC plan, as it was then called, was a carefully conceived blueprint of the research and engineering needed to leapfrog the U.S. computer industry and destroy its world supremacy. Worse, even though American universities had produced the basic research the Japanese would rely on, American companies were as serenely unaware of danger as the battleships that swung at anchor in Pearl Harbor more than 40 years ago. "My good friends in the U.S. corporations were deeply asleep," Dertouzos says.

Today the battle lines are drawn. Armed with fresh commitments of money and manpower, the U.S. is taking on Japan for control of the advances technologies that will dominate computing in the late 1980s and the 1990s. Nor is this simply a struggle for an industry: entire economies will be reshaped by the coming radical changes in information processing. Supercomputer speed is already being used commercially for aircraft design, oil, and mineral exploration, weather forecasting and computer circuit design—all of which require vast amounts of calculation. Supercomputers may soon be put to work in the automobile and shipbuilding industries, and pressed into the service of genetic engineers and economic forecasters.

The supersecret National Security Agency, the cryptographers' "Puzzle Palace," is already a heavy user of today's supercomputers. And the Pentagon's futuristic laser

weapons systems based in space will depend on supercomputers. "This assault is far more serious to our future than the automobiles sold from Japan, because the computer is at the root of every major future change," Dertouzos warns. "The Japanese recognize that whoever controls the information revolution has, in effect, some form of increased geopolitical control."

Until now the U.S. has dominated advanced computer technologies: the world's supercomputers have all been American-made. There are 74 in operation, and they are very powerful machines, capable of performing several hundred million operations per second. They are so fast and their electronic circuitry is so dense that giant refrigeration units must pump a freon-gas coolant through the machines just to keep them from melting down.

Yet the current supercomputers are only at the threshold of what computer designers think can be achieved; the next generation of advanced supercomputers will make today's machines look like handheld calculators. "We have problems that would take 500 to 1,000 hours to solve [on today's supercomputers]," says David Nowak, division leader for computational physics at Lawrence Livermore National Laboratory, where a cluster of seven supercomputers—known as "Octopus"—is used for nuclear weapons research. Before the end of the century, computer scientists hope to develop machines that not only crunch numbers at high speed but also exhibit artificial intelligence—computers that can think and reason somewhat like human beings and that can understand information conveyed by sight, speech, and motion.

The question is, which nation's scientists will get there first? . . .

The Pentagon's Defense Advanced Research Projects Agency (DARPA) is more than any other single agency in the world, responsible for the shape of advanced computer science today—and for many technologies now in widespread commercial use. Over the past 20 years, DARPA has poured half a billion dollars into computer research, in the process virtually creating the science of artificial intelligence. The first supercomputer, built in 1964, was a DARPA project. Computer time sharing, a fundamental advance, came out of work sponsored by DARPA; so did packet-switched networks, the workhorses of today's telecommunications data networks. And computer graphics—now used on desktop computers and video-arcade screens as well as in F-10 cockpits—is a DARPA-sponsored invention.

DARPA's next priority is a push for advanced supercomputing and artificial intelligence technologies that may cost as much as $1 billion. DARPA plans to do everything the Japanese have set out to accomplish—and more.

"Dual-Use" Technology, 1991

JEFF BINGAMAN AND BOBBY R. INMAN

The collapse of the Soviet Union overturned the assumptions upon which United States military security rested for 40 years. When President Bush was asked to describe the threat to which defense planning should now be directed, he responded, airplane crash, computers would have been seen first as text-handling and picture-making machines, and only later developed for mathematics and business.)

We would call it the All-Purpose Machine here, except that for historical reasons it has been slapped with the other name.

But that doesn't mean it has a fixed way of operating. On the contrary.

COMPUTERS HAVE NO NATURE AND NO CHARACTER,

save that which has been put into them by whoever is creating the program for a particular purpose. Computers are, unlike any other piece of equipment, perfectly BLANK. And that is how we have projected on it so many different faces. . . .

Computer people may best be thought of as a new ethnic group, very much unto themselves. Now, it is very hard to characterize ethnic groups in words, and certain to give offense, but if I had to choose one word for them it would be *elfin*. . . .

One common trait of our times—the technique of obscuring oneself—may be more common among computer people than others. . . . Perhaps a certain disgruntlement with the world of people fuses with fascination for (and envy of?) machines. Anyway, many of us who have gotten along badly with people find here a realm of abstractions to invent and choreograph, privately and with continuing control. A strange house for the emotions, this. Like Hegel, who became most eloquent and ardent when he was lecturing at his most theoretical, it is interesting to be among computer freaks boisterously explaining the cross-tangled ramifications of some system they have seen or would like to build.

(A syndrome to ponder. I have seen it more than once: the technical person who, with someone he cares about, cannot stop talking about his ideas for a project. A poignant type of Freudian displacement.)

A sad aspect of this, incidentally, is by no means obvious. This is that the same computer folks who chatter eloquently about systems that fascinate them tend to fall dark and silent while someone *else* is expounding his own fascinations. You would expect that the person with effulgent technical enthusiasms would really click with kindred spirits. In my experience this only happens briefly: hostilities and disagreements boil out of nowhere to cut the good mood. My only conclusion is that the same spirit that originally drives us muttering into the clockwork feels threatened when others start monkeying with what has been controlled and private fantasy.

This can be summed up as follows: NOBODY WANTS TO HEAR ABOUT ANOTHER GUY'S SYSTEM.

The Shaping of the Personal Computer

MARTIN CAMPBELL-KELLY AND WILLIAM ASPRAY

The enabling technology for the personal computer, the microprocessor, was developed during 1969 to 1971 in the semiconductor firm Intel. . . .

When Intel first began operations in 1968, it specialized in the manufacture of semiconductor memory and custom-designed chips. Intel's custom-chip sets were

typically used in calculators, video games, electronic test gear, and control equipment. In 1969 Intel was approached by the Japanese calculator manufacturer Busicom to develop a chip set for a new scientific calculator—a fairly up-market model that would include trigonometrical and other advanced mathematical functions. The job of designing the chip set was assigned to Ted Hoff.

Hoff decided that instead of specifically designed logic chips for the calculator, a better approach would be to design a general-purpose chip that could be programmed with the specific calculator functions. Such a chip would of course be a rudimentary computer in its own right, although it was some time before the significance of this dawned inside Intel.

The new calculator chip, known as the 4004, was delivered to Busicom in early 1971. . . .

It would have been technically possible to produce an affordable personal computer (costing less than $2,000, say) anytime after the launch of the 4004, in November 1971. But it was not until nearly six years later that a real consumer product emerged, in the shape of the Apple II. The long gestation of the personal computer contradicts the received wisdom of its having arrived almost overnight. It was rather like the transition from wireless telegraphy to radio broadcasting, which the newspapers in 1921 saw as a "fad" that "seemed to come from nowhere"; in fact, it took several years, and the role of the hobbyist was crucial. . . .

The computer hobbyist was typically a young male technophile. Most hobbyists had some professional competence. If not working with computers directly, they were often employed as technicians or engineers in the electronics industry. This typical hobbyist had cut his teeth in his early teens on electronic construction kits, bought through mail-order advertisements in one of the popular electronics magazines. Many of the hobbyists were active radio amateurs. But even those who were not radio amateurs owed much to the "ham" culture, which descended in an unbroken line from the early days of radio. After World War II, radio amateurs and electronic hobbyists moved on to building television sets and hi-fi kits advertised in magazines such as *Popular Electronics* and *Radio Electronics*. In the 1970s, the hobbyists lighted on the computer as the next electronic bandwagon.

Their enthusiasm for computing had often been produced by the hands-on experience of using a minicomputer at work or in college. The dedicated hobbyist hungered for a computer at home for recreational use, so that he could explore its inner complexity, experiment with computer games, and hook it up to other electronic gadgets. However, the cost of a minicomputer—typically $20,000 for a complete installation—was way beyond the pocket of the average hobbyist. To the nonhobbyist, why anyone would have wanted his own computer was a mystery: It was sheer techno-enthusiasm, and one can no more explain it than one can explain why people wanted to build radio sets sixty years earlier when there were no broadcasting stations.

It is important to understand that the hobbyist could conceive of hobby computing only in terms of the technology with which he was familiar. This was not the personal computer as we know it today; rather, the computing that the hobbyist had in mind in the early 1970s was a minicomputer hooked up to a teletype equipped

most expensive part of the minicomputer—the central processing unit—remained much too costly for the amateur. The allure of the microprocessor was that it would reduce the price of the central processor by vastly reducing the chip count in the conventional computer.

The amateur computer culture was widespread. While it was particularly strong in Silicon Valley and around Route 128 [in Massachusetts], computer hobbyists were to be found all over the country. The computer hobbyist was primarily interested in tinkering with computer hardware; software and applications were very much secondary issues.

Fortunately, the somewhat technologically fixated vision of the computer hobbyists was leavened by a second group of actors: the advocates of "computer liberation." It would, perhaps, be overstating the case to describe computer liberation as a movement, but there was unquestionably a widely held desire to bring computing to ordinary people. Computer liberation was particularly strong in California, and this perhaps explains why the personal computer was developed in California rather than (say) around Route 128.

Computer liberation sprang from a general malaise in the under-thirty crowd in the post-Beatles, post–Vietnam War period of the early 1970s. There was still a strong anti-establishment culture that expressed itself through the phenomena of college dropouts and campus riots, communal living, hippie culture, and alternative lifestyles sometimes associated with drugs. Such a movement for liberation would typically want to wrest communications technologies from vested corporate interests. In an earlier generation the liberators might have wanted to appropriate the press, but in fact the technology of printing and distribution channels were freely available, so that the young, liberal-minded community was readily able to communicate through magazines such as *Rolling Stone* as well as a vast underground press. On the other hand, computer technology was unquestionably not freely available; it was mostly rigidly controlled in government bureaucracies or private corporations. The much vaunted computer utility was, at $10 to $20 per hour, beyond the reach of ordinary users.

The most articulate spokesperson for the computer liberation idea was Ted Nelson, the financially independent son of the Hollywood actress Celeste Holm. Among Nelson's radical visions of computing was an idea called hypertext, which he first described in the mid-1960s. *Hypertext* was a system by which an untrained person could navigate through a universe of information held on computers. Before such an idea could become a reality, however, it was necessary to "liberate" computing: to make it accessible to ordinary people at a trivial cost. In the 1970s Nelson promoted computer liberation as a regular speaker at computer hobbyist gatherings. He took the idea further in his self-published books *Computer Liberation* and *Dream Machines*, which appeared in 1974. While Nelson's uncompromising views and his unwillingness to publish his books through conventional channels perhaps added to his anti-establishment appeal, this created a barrier between himself and the academic and commercial establishments.

. . . Nelson influenced mainly the young, predominantly male, local Californian technical community.

It is important to understand that personal computing in 1974, whether it was the vision of computer liberation or that of the computer hobbyist, bore little resemblance

a self-contained machine, somewhat like a typewriter, with a keyboard and screen, an internal microprocessor-based computing engine, and a floppy disk for long-term data storage. In 1974 the computer-liberation vision of personal computing was a terminal attached to a large, information-rich computer utility at very low cost, while the computer hobbyist's vision was that of a traditional minicomputer. What brought together these two groups, with such different perspectives, was the arrival of the first hobby computer, the Altair 8800. . . .

In January 1975 the first microprocessor-based computer, the Altair 8800, was announced on the front cover of *Popular Electronics*. The Altair 8800 is often described as the first personal computer. This was true only in the sense that its price was so low that it could be realistically bought by an individual. In every other sense the Altair 8800 was a traditional minicomputer. Indeed, the blurb on the front cover of *Popular Electronics* described it as exactly that: "Exclusive! Altair 8800. The most powerful minicomputer project ever presented—can be built for under $400."

The Altair 8800 closely followed the marketing model of the electronic hobbyist kit: It was inexpensive ($397) and was sold by mail order as a kit that the enthusiast had to assemble himself. In the tradition of the electronics hobbyist kit, the Altair 8800 often did not work when the enthusiast had constructed it; and even if it did work, it did not do anything very useful. The computer consisted of a single box containing the central processor, with a panel of switches and neon bulbs on the front; it had no display, no keyboard, and not enough memory to do anything useful. Moreover, there was no way to attach a device such as a teletype to the machine to turn it into a useful computer system.

The only way the Altair 8800 could be programmed was by entering programs in pure binary code by flicking the hand switches on the front. When loaded, the program would run; but the only evidence of its execution was the change in the shifting pattern of the neon bulbs on the front. This limited the Altair 8800 to programs that only a dedicated computer hobbyist would ever be able to appreciate. Entering the program was extraordinarily tedious, taking several minutes—but as there were only 256 bytes of memory, there was a limit to the complexity of programs that could be attempted.

The Altair 8800 was produced by a tiny Albuquerque, New Mexico, electronics kit supplier, Micro Instrumentation Telemetry Systems (MITS). The firm had originally been set up by an electronics hobbyist, Ed Roberts, to produce radio kits for model airplanes. . . . The Altair 8800 was unprecedented and in no sense a "rational" product—it would appeal only to an electronics hobbyist of the most dedicated kind, and even that was not guaranteed.

Despite its many shortcomings, the Altair 8800 was the grit around which the pearl of the personal-computer industry grew during the next two years. The limitations of the Altair 8800 created the opportunity for small-time entrepreneurs to develop "add-on" boards so that extra memory, conventional teletypes, and audio-cassette recorders (for permanent data storage) could be added to the basic machine. Almost all of these start-up companies consisted of two or three people—mostly computer hobbyists hoping to turn their pastime to profit. A few other entrepreneurs developed software for the Altair 8800.

almost without parallel, his background was quite typical of a 1970s software nerd—a term that conjures up an image of a pale, male adolescent, lacking in social skills, programming by night and sleeping by day, oblivious to the wider world and the need to gain qualifications and build a career. This stereotype, though exaggerated, contains an essential truth; nor was it a new phenomenon—the programmer-by-night has existed since the 1950s. Indeed, programming the first personal computers had many similarities to programming a 1950s mainframe: There were no advanced software tools, and programs had to be hand-crafted in the machine's own binary codes so that every byte of the tiny memory could be used to its best advantage.

Gates, born in 1955 in Seattle to upper-middle-class parents, was first exposed to computers in 1969, when he learned to program in BASIC using a commercial time-sharing system on which his high-school rented time. He and his close friend, Paul Allen, two years his senior, discovered a mutual passion for programming. . . .

The launch of the Altair 8800 in 1975 transformed Gate's and Allen's lives. Almost as soon as they heard of the machine, they recognized the software opportunity it represented and proposed to MITS' Ed Roberts that they should develop a BASIC programming system for the new machine. . . .

Gates and Allen formed a partnership they named Micro-Soft (the hyphen was later dropped), and after six weeks of intense programming effort they delivered a BASIC programming system to MITS in February 1975. . . . Gates abandoned his formal education. During the next two years, literally hundreds of small firms entered the microcomputer software business, and Microsoft was by no means the most prominent.

The Altair 8800, and the add-on boards and software that were soon available for it, transformed hobby electronics in a way not seen since the heyday of radio. In the spring of 1975, for example, the "Homebrew Computer Club" was established in Menlo Park, on the edge of Silicon Valley. Besides acting as a swap shop for computer components and programming tips, it also provided a forum for the computer-hobbyist and computer-liberation cultures to meld. . . .

While it had taken the mainframe a decade to be transformed from laboratory instrument to business machine, the personal computer was transformed in just two years. The reason for this rapid development was that most of the subsystems required to create a personal computer already existed: keyboards, screens, disk drives, and printers. It was just a matter of putting the pieces together. Hundreds of firms—not just on the West Coast, but all over the country—sprang up over this two-year period. They were mostly tiny start-ups, consisting of a few computer hobbyists or young computer professionals; they supplied complete computers, add-on boards, peripherals, or software. Within months of its initial launch at the beginning of 1975, the Altair 8800 had itself been eclipsed by dozens of new models produced by firms such as Applied Computer Technology, IMSAI, North Star, Cromemco, and Vector. . . .

Most of the new computer firms fell almost as quickly as they rose, and only a few survived beyond the mid-1980s. Apple computer was the rare exception in that it made it into the Fortune 500 and achieved long-term global success. Its initial trajectory, however, was quite typical of the early hobbyist start-ups.

Apple was founded by two young computer hobbyists, Stephen Wozniak and Steve Jobs. Wozniak grew up in Cupertino, California, in the heart of the booming West

air they breathed. Wozniak took to electronics almost as soon as he could think abstractly; he was a talented hands-on engineer, lacking any desire for a deeper, academic understanding. He obtained a radio amateur operating license while in sixth grade. . . .

While Wozniak was a typical, if unusually gifted, hobbyist, Steve Jobs bridged the cultural divide between computer hobbyism and computer liberation. That Apple Computer ultimately became a global player in the computer industry is largely due to Jobs's evangelizing of the personal computer, his ability to harness Wozniak's engineering talent, and his willingness to seek out the organizational capabilities needed to build a business.

Born in 1955, Jobs was brought up by adoptive blue-collar parents. Although not a child of the professional electronics-engineering classes, Jobs took to the electronic hobbyism that he saw all around him. While a capable enough engineer, he was not in the same league as Wozniak. . . .

Something of a loner, and not academically motivated, Jobs drifted in and out of college in the early 1970s before finding a well-paid niche as a games designer for Atari. An admirer of the Beatles, like them Jobs spent a year pursuing transcendental meditation in India and turned vegetarian. Jobs and Wozniak made a startling contrast: Wozniak was the archetypal electronic hobbyist with social skills to match, while Jobs affected an aura of inner wisdom, wore open-toed sandals, had long, lank hair, and sported a Ho Chi Minh beard.

The turning point for both Jobs and Wozniak was attending the Homebrew Computer Club in early 1975. Although Wozniak knew about microprocessors from his familiarity with the calculator industry, he had not up to that point realized that they could be used to build general-purpose computers and had not heard of the Altair 8800. But he had actually built a computer, which was more than could be said of most Homebrew members at that date, and he found himself among an appreciative audience. He quickly took up the new microprocessor technology, and within a few weeks had thrown together a computer based on the Mostek 6502 chip. He and Jobs called it the "Apple," for reasons that are now lost in time, but possibly for the Beatles' record label.

While Jobs never cared for the "nit-picking technical debates" of the Homebrew computer enthusiasts, he did recognize the latent market they represented. He therefore cajoled Wozniak into developing the Apple computer and marketing it, initially through the Byte Shop. The Apple was a very crude machine, consisting basically of a naked circuit board, lacking a case, a keyboard, or screen, or even a power supply. Eventually about two hundred were sold, each hand-assembled by Jobs and Wozniak in the garage of Job's parents.

In 1976 Apple was just one of dozens of computer firms competing for the dollars of the computer hobbyist. Jobs recognized before most, however, that the microcomputer had the potential to be a consumer product for a much broader market if it were appropriately packaged. To be a success as a product, the microcomputer would have to be presented as a self-contained unit in a plastic case, able to be plugged into a standard household outlet just like any other appliance; it would need a keyboard to enter data, a screen to view the results of a computation, and some form of long-term storage to hold data and programs. Most important, the machine

in a nutshell, was the specification for the Apple II that Jobs passed down to Wozniak to create.

For all his naïveté as an entrepreneur Jobs understood, where few of his contemporaries did, that if Apple was to become a successful company, it would need access to capital, professional management, public relations, and distribution channels. None of these was easy to find at a time when the personal computer was unknown outside hobbyist circles. Jobs's evangelizing was called on in full measure to acquire these capabilities. During 1976, while Wozniak designed the Apple II, Jobs secured venture capital from Mike Markkula, to whom he had been introduced by his former employer at Atari, Nolan Bushnell. Markkula was a thirty-four-year-old former Intel executive who had become independently wealthy from stock options. Through Markkula's contacts, Jobs located an experienced young professional manager from the semiconductor industry, Mike Scott, who agreed to serve as president of the company. Scott would take care of operational management, leaving Jobs free to evangelize and determine the strategic direction of Apple. The last piece of Jobs's plan fell into place when he persuaded the prominent public relations company Regis McKenna to take on Apple as a client.

Throughout 1976 and early 1977, while the Apple II was perfected, Apple Computer remained a tiny company with fewer than a dozen employees occupying 2,000 square feet of space in Cupertino, California. . . .

During 1977 three distinct paradigms for the personal computer emerged, represented by three leading manufacturers: Apple, Commodore Business Machines, and Tandy, each of which defined the personal computer in terms of its own existing culture and corporate outlooks.

If there can be said to be a single moment when the personal computer arrived in the public consciousness, then it was at the West Coast Computer Faire in April 1977, when the first two machines for the mass consumer, the Apple II and the Commodore PET, were launched. Both machines were instant hits, and for a while they vied for market leadership. At first glance the Commodore PET looked very much like the Apple II in that it was a self-contained appliance with a keyboard, a screen, and a cassette tape for program storage, and with BASIC ready-loaded so that users could write programs.

The Commodore PET, however, coming from Commodore Business Machines—a firm that had originally made electronic calculators—was not so much a computer as a calculator writ large. For example, the keyboard had the tiny buttons of a calculator keypad rather then the keyboard of a standard computer terminal. Moreover, like a calculator, the PET was a closed system, with no potential for add-ons such as printers or floppy disks. Nevertheless, this narrow specification and the machine's low price appealed to the educational market, where it found a niche supporting elementary computer studies and BASIC programming; eventually several hundred thousand machines were sold.

. . . The Apple II was . . . far more appealing to the computer hobbyist because it offered the opportunity to engage with the machine by customizing it and using it for novel applications that the inventors could not envisage.

In August 1977 the third major computer vendor, Tandy, entered the market,

mainly of electronic hobbyists and buyers of video games. The low price was achieved by the user having to use a television set for a screen and an audiocassette recorder for program storage. The resulting hook-up was no hardship to the typical Tandy customer, although it would have been out of place in an office.

Thus, by the fall of 1977, although the personal computer had been defined physically as an artifact, a single constituency had not yet been established. For Commodore the personal computer was seen as a natural evolution of its existing calculator line. For Tandy it was an extension of its existing electronic-hobbyist and video games business. For Apple the machine was initially aimed at the computer hobbyist.

Jobs's ambition and vision went beyond the hobby market, and he envisioned the machine also being used as an appliance in the home—perhaps the result of his experience as a designer of domestic video games. This ambiguity was revealed by the official description of the Apple II as a "home/personal computer." The advertisement that Regis McKenna produced to launch the Apple II showed a housewife doing kitchen chores, while in the background her husband sat at the kitchen table hunched over an Apple II, seemingly managing the household's information. The copy read:

> The home computer that's ready to work, play and grow with you. . . . You'll be able to organize, index and store data on household finances, income taxes, recipes, your biorhythms, balance your checking account, even control your home environment.

These domestic projections for the personal computer were reminiscent of those for the computer utility in the 1960s, and were equally misguided. Moreover, the advertisement did not point out that these domestic applications were pure fantasy—there was no software available for "biorhythms," accounts, or anything else.

The constituency for the personal computer would be defined by the software that was eventually created for it. . . .

The biggest market, initially, was for games software, which reflected the existing hobbyist customer base:

> When customers walked into computer stores in 1979, they saw racks of software, wall displays of software, and glass display cases of software. Most of it was games. Many of these were outer space games—*Space, Space II, Star Trek.* Many games appeared for the Apple, including Programma's simulation of a video game called *Apple Invaders.* Companies such as Muse, Sirius, Broderbund, and On-Line systems reaped great profits from games.

Computer games are often overlooked in discussions of the personal-computer software industry, but they played an important role in its early development. Programming computer games created a corps of young programmers who were very sensitive to what we now call human/computer interaction. The most successful games were ones that needed no manuals and gave instant feedback. . . .

The market of packaged software for business applications developed between 1978 and 1980, when three generic applications enabled the personal computer to become an effective business machine: the spreadsheet, the word processor, and the database. All these types of software already existed in the ordinary mainframe com-

The first application to receive wide acceptance was the VisiCalc spreadsheet. The originator of VisiCalc was a twenty-six-year-old Harvard MBA student, Daniel Bricklin. . . .

Bricklin's program used about 25,000 bytes of memory, which was about as big as a personal computer of the period could hold, but was decidedly modest by mainframe standards. The personal computer, however, offered some significant advantages that were not obvious at the outset. Because the personal computer was a stand-alone, self-contained system, changes to a financial model were displayed almost instantaneously compared with the minute or so it would have taken on a conventional computer. This fast response enabled a manager to explore a financial model with great flexibility, asking what were later known as "what if?" questions. It was almost like a computer game for executives.

When it was launched in December 1979, VisiCalc was an overnight success. Not only was the program a breakthrough as a financial tool but its users experienced for the first time the psychological freedom of having a machine of one's own, on one's desk, instead of having to accept the often mediocre take-it-or-leave-it services of a computer center. Moreover, at $3,000, including software, it was possible to buy an Apple II and VisiCalc out of a departmental, or even a personal, budget. . . .

Word processing on personal computers did not develop until about 1980. . . .

The first successful firm to produce word-processing software was MicroPro, founded by the entrepreneur Seymour Rubinstein in 1978. Rubinstein, then in his early forties, was formerly a mainframe software developer. He had a hobbyist interest in amateur radio and electronics, however, and when the first microcomputer kits became available, he bought one. He recognized very early on the personal computer's potential as a word processor. . . .

During 1980, with dozens of spreadsheet and word-processing packages on the market and the launch of the first database products, the potential of the personal computer as an office machine became clearly recognizable. At this point the traditional business machine manufacturers, such as IBM, began to take an interest. . . .

IBM was not, in fact, the giant that slept soundly during the personal-computer revolution. IBM had a sophisticated market research organization that attempted to predict market trends. The company was well aware of microprocessors and personal computers. Indeed, in 1975 it had developed a desktop computer for the scientific market (the model 5100), but it did not sell well. . . .

Once the personal computer became clearly defined as a business machine in 1980, IBM reacted with surprising speed. The proposal that IBM should enter the personal-computer business came from William C. Lowe, a senior manager who headed the company's "entry-level systems" in Boca Raton, Florida. . . .

For nearly a century IBM had operated a bureaucratic development process by which it typically took three years for a new product to reach the market. Part of the delay was due to IBM's century-old vertical integration practice, by which it maximized profits by manufacturing in-house all the components used in its products: semiconductors, switches, plastic cases, and so on. Lowe argued that IBM should instead adopt the practice of the rest of the industry by outsourcing all the components it did not already have in production, including software. Lowe proposed yet

Surprisingly, in light of its stuffy image, IBM's top management agreed to all that Lowe recommended, and within two weeks of his presentation he was authorized to go ahead and build a prototype, which had to be ready for the market within twelve months. The development of the personal computer would be known internally as Project Chess.

IBM's relatively late entry into the personal computer market gave it some significant advantages. First, it could make use of the second generation of microprocessors (which processed sixteen bits of data at a time instead of eight); this would make the IBM personal computer significantly faster than any other machine on the market. IBM chose to use the Intel 8088 chip, thereby guaranteeing Intel's future prosperity.

Although IBM was the world's largest software developer, paradoxically it did not have the skills to develop software for personal computers. Its bureaucratic software development procedures were slow and methodical, and geared to large software artifacts: the company lacked the critical skills needed to develop the "quick-and-dirty" software needed for personal computers.

IBM initially approached Gary Kildall of Data Research—the developer of the CP/M operating system—for operating software for the new computer, and herein lies one of the more poignant stories in the history of the personal computer. For reasons now muddied, Kildall blew the opportunity. One version of the story has it that he refused to sign IBM's nondisclosure agreement, while another version has him doing some recreational flying while the dark-suited IBMers cooled their heels below. In any event, the opportunity passed Digital Research by and moved on to Microsoft. Over the next decade, buoyed by the revenues from its operating system for the IBM personal computer, Microsoft became the quintessential business success story of the late twentieth century, and Gates became a billionaire at the age of thirty-one. Hence, for all of Gates's self-confidence and remarkable business acumen, he owes almost everything to being in the right place at the right time.

The IBM entourage arrived at Bill Gates and Paul Allen's Microsoft headquarters in July 1980. It was then a tiny (thirty-two-person) company located in rented offices in downtown Seattle. It is said that Gates and Allen were so keen to win the IBM contract that they actually wore business suits and ties. Although Gates may have appeared a somewhat nerdish twenty-nine-year-old who looked fifteen, he came from an impeccable background, was palpably serious, and showed a positive eagerness to accommodate the IBM culture. For IBM, he represented as low a risk as any of the personal-computer software firms, almost all of which were noted for their studied contempt for Big Blue. It is said that when John Opel, IBM's president, heard about the Microsoft deal, he said, "Is he Mary Gates's son?" He was. Opel and Gates's mother both served on the board of the United Way.

At the time that Microsoft made its agreement with IBM for an operating system, it did not have an actual product, nor did it have the resources to develop one in IBM's time scale. However, Gates obtained a suitable piece of software from a local software firm, Seattle Computer Products, for $30,000 cash. Eventually, the operating system, known as MS-DOS, would be bundled with almost every IBM personal computer and compatible machine, earning Microsoft a royalty of between $10 and $50 on every copy sold. . . .

Early in 1981, only six months after the inception of Project Chess, IBM

campaign. Market research suggested that the personal computer still lay in the gray area between regular business equipment and a home machine. The advertising campaign was therefore ambiguously aimed at both the business and home user. The machine was astutely named the IBM Personal Computer, suggesting that the IBM machine and the personal computer were synonymous. For the business user, the fact that the machine bore the IBM logo was sufficient to legitimate it inside the corporation. For the home user, however, market research revealed that although the personal computer was perceived as a good thing, it was also seen as intimidating—and IBM itself was seen as "cold and aloof." The Chiat Day campaign attempted to allay these fears by featuring in its advertisements a Charlie Chaplin lookalike and alluding to Chaplin's famous movie *Modern Times*. Set in a futuristic automated factory, *Modern Times* showed the "little man" caught up in a world of hostile technology, confronting it, and eventually overcoming it. The Charlie Chaplin figure reduced the intimidation factor and gave IBM "a human face." . . .

The IBM Personal Computer was given its press launch in New York on 12 August. There was intense media interest, which generated many headlines in the computer and business press. In the next few weeks the IBM Personal Computer became a runaway success that exceeded almost everyone's expectations, inside and outside the company. While many business users had hesitated over whether to buy an Apple or a Commodore or a Tandy machine, the presence of the IBM logo convinced them that the technology was for real: IBM had legitimated the personal computer. There was such a demand for the machine that production could not keep pace, and retailers could do no more than placate their customers by placing their names on a waiting list. Within days of the launch, IBM decided to quadruple production.

During 1982–83 the IBM Personal Computer became an industry standard. Most of the popular software packages were converted to run on the machine, and the existence of this software reinforced its popularity. This encouraged other manufacturers to produce "clone" machines, which ran the same software.

The Second Self, 1982

SHERRY TURKLE

The subjects of my study are men and women who bought personal computer systems in the four years that followed the 1975 announcement of the "Altair"—the first computer small enough to sit on a desktop, powerful enough to support high-level language programming, and that you could build for only $420. My study began in 1978 with a questionnaire survey answered by 95 New England computer hobbyists (their names had been drawn from the roster of a home computer club and from the subscription list of a personal computer magazine), and continued during 1978 and 1979 with nearly 300 hours of conversation with 50 individuals who owned home computers. What I found can be read historically: a study of the pioneer users of an increasingly ubiquitous technology. But most central to the intent of this essay is to use the story of the early hobbyists as a window into the highly personal ways in

which individuals appropriate technologies. It is a case study of the "subjective computer," the computer as a material for thinking, for feeling, for "working through."

My emphasis on the subjective is at odds with a widespread ideology that quickly grew up around the emergent computer hobbyist culture. The Altair, aimed at a strictly hobby market, was followed by other small systems—the Pet, the Sol, and, most successfully, the Radio Shack TRS-80 and the Apple, marketed to less specialized audiences of small businessmen and curious householders. With this explosion of hardware came a lot of rhetoric about a personal computer revolution. Most of the talk, both from the companies that marketed the machines and from those who claimed to be the most visionary spokesmen for the people who bought them, was about all of the things that a home computer could do for you. The utilitarian, "genie in the bottle," ideology is expressed in the content of hobbyist conventions and magazines, filled with articles on how to make your home computer dim your lights, control your thermostat, run an inventory system for your kitchen or toolroom. And it is also found in writing on the personal computer from outside the hobbyist world. . . .

This instrumental view is an important ingredient of the hobbyist ideology but it is not the whole story. In the course of my work I found a very different answer from within. Most hobbyists do make their computers "strut their stuff," but their sense of engagement and energy are found primarily in the non-instrumental uses of the technology. When asked in a questionnaire "What first attracted you to computers?" more than half the respondents gave reasons that were highly subjective. In response to an open-ended question, 26 percent said that they were first attracted to computers by an appeal that was intellectual, aesthetic, involved with the fun of what I would call "cognitive play." They wrote of "puzzle solving," of "the elegance of using computer techniques to handle problems," of the "beauty of understanding a system of many levels of complexity." They described what they did with their home computers with metaphors like "mind stretching" and "using the computer's software to understand my wetware." Another 26 percent wrote of reasons for getting involved that seemed more emotional than intellectual. They wrote of the "ego boost" or "sense of power" that comes from knowing how to run a computer, of the "prestige of being a pioneer in a developing field," of the "feeling of control when I work in a safe environment of my own creation."

The hobbyists who responded to my survey seemed familiar with . . . people who ask them what they do with their computers and who won't take "cognitive play" for an answer. David, a 19-year-old undergraduate at a small engineering school, put it this way: "People come over and see my computer and they look at it, then they look at me, then they ask me what useful thing I do with it, like does it wash floors, clean laundry or do my income tax—when I respond no, they lose interest." David said that when he started out, he was attracted to the computer because "I like the idea of making a pile of hardware do something useful, like doing real time data processing . . . like picking up morse code with an amateur radio and transcribing it automatically into text," but in his list of things that he currently does with his computer, an instrumental discourse is most notable for its absence:

> Conway's GAME OF LIFE in assembly code was a challenge, forced me to "think logically," and gave the pleasure of making something work the way I wanted it to . . . Having

Cyberspace, 1984

WILLIAM GIBSON

"The matrix has its roots in primitive arcade games," said the voice-over, "in early graphics programs and military experimentation with cranial jacks." On the Sony, a two-dimensional space war faded behind a forest of mathematically generated ferns, demonstrating the spacial possibilities of logarithmic spirals; cold blue military footage burned through, lab animals wired into test systems, helmets feeding into fire control circuits of tanks and war planes. "Cyberspace. A consensual hallucination experienced daily by billions of legitimate operators, in every nation, by children being taught mathematical concepts . . . A graphic representation of data abstracted from the banks of every computer in the human system. Unthinkable complexity. Lines of light ranged in the nonspace of the mind, clusters and constellations of data. Like city lights, receding. . . ."

. . . He settled the black terry sweatband across his forehead, careful not to disturb the flat Sendai dermatrodes. He stared at the deck on his lap, not really seeing it. . . .

He closed his eyes.

Found the ridged face of the power stud.

And in the bloodlit dark behind his eyes, silver phosphenes boiling in from the edge of space, hypnagogic images jerking past like film compiled from random frames. Symbols, figures, faces, a blurred, fragmented mandala of visual information.

Please, he prayed, *now*—

A gray disk, the color of Chiba sky.

Now—

Disk beginning to rotate, faster, becoming a sphere of paler gray. Expanding—

And flowed, flowered for him, fluid neon origami trick, the unfolding of his distanceless home, his country, transparent 3D chessboard extending to infinity. Inner eye opening to the stepped scarlet pyramid of the Eastern Seaboard Fission Authority burning beyond the green cubes of Mitsubishi Bank of America, and high and very far away he saw the spiral arms of military systems, forever beyond his reach.

And somewhere he was laughing, in a white-painted loft, distant fingers caressing the deck, tears of release streaking his face.

A Cyborg Manifesto

DONNA HARAWAY

This chapter is an effort to build an ironic political myth faithful to feminism, socialism, and materialism. Perhaps more faithful as blasphemy is faithful, than as reverent worship and identification. Blasphemy has always seemed to require taking things very seriously. I know no better stance to adopt from within the secular-religious, evangelical traditions of United States politics, including the politics of socialist feminism.

From William Gibson, *Neuromancer* (New York: Ace Books, 1984), pp. 51–52.

From Donna Haraway, *Simians, Cyborgs, and Women* (London: Free Association Books, 1991), pp. 149–

Blasphemy protects one from the moral majority within, while still insisting on the need for community. Blasphemy is not apostasy. Irony is about contradictions that do not resolve into larger wholes, even dialectically, about the tension of holding incompatible things together because both or all are necessary and true. Irony is about humour and serious play. It is also, a rhetorical strategy and a political method, one I would like to see more honoured within socialist-feminism. At the centre of my ironic faith, my blasphemy, is the image of the cyborg.

A cyborg is a cybernetic organism, a hybrid of machine and organism, a creature of social reality as well as a creature of fiction. Social reality is lived social relations, our most important political construction, a world-changing fiction. The international women's movements have constructed "women's experience," as well as uncovered or discovered this crucial collective object. This experience is a fiction and fact of the most crucial, political kind. Liberation rests on the construction of the consciousness, the imaginative apprehension, of oppression, and so of possibility. The cyborg is a matter of fiction and lived experience that changes what counts as women's experience in the late twentieth century. This is a struggle over life and death, but the boundary between science and social reality is an optical illusion.

Contemporary science fiction is full of cyborgs—creatures simultaneously animal and machine, who populate worlds ambiguously natural and crafted. Modern medicine is also full of cyborgs, of couplings between organism and machine, each conceived as coded devices, in an intimacy and with a power that was not generated in the history of sexuality. . . . Modern production seems like a dream of cyborg colonization work, a dream that makes the nightmare of Taylorism seem idyllic. And modern war is a cyborg orgy, coded by C^3I, command-control-communication-intelligence, an $84 billion item in 1984's US defense budget. . . .

By the late twentieth century, our time, a mythic time, we are all chimeras, theorized and fabricated hybrids of machine and organism; in short, we are cyborgs. The cyborg is our ontology; it gives us our politics. The cyborg is a condensed image of both imagination and material reality, the two joined centres structuring any possibility of historical transformation. In the traditions of "Western" science and politics—the tradition of racist male-dominant capitalism; the tradition of progress; the tradition of the appropriation of nature as resource for the productions of culture; the tradition of reproduction of the self from the reflections of the other—the relation between organism and machine has been a border war. The stakes in the border war have been the territories of production, reproduction, and imagination. This chapter is an argument for *pleasure* in the confusion of boundaries and for *responsibility* in their construction. It is also an effort to contribute to socialist-feminist culture and theory in a postmodernist, non-naturalist mode and in the utopian tradition of imagining a world without gender, which is perhaps a world without genesis, but maybe also a world without end. The cyborg incarnation is outside salvation history. . . .

The cyborg is a creature in a post-gender world.

. . . In a sense, the cyborg has no origin story in the Western sense—a "final" irony since the cyborg is also the awful apocalyptic *telos* of the "West's" escalating dominations of abstract individuation, an ultimate self united at last from all dependency, a man in space. An origin story in the "Western," humanist sense depends on the myth of original unity, fullness, bliss and terror, represented by the phallic

From ARPANET to Internet

BRUCE STERLING

Some thirty years ago, the RAND Corporation, America's foremost Cold War think-tank, faced a strange strategic problem. How could the U.S. authorities successfully communicate after a nuclear war?

Postnuclear America would need a command-and-control network, linked from city to city, state to state, base to base. But no matter how thoroughly that network was armored or protected, its switches and wiring would always be vulnerable to the impact of atomic bombs. A nuclear attack would reduce any conceivable network to tatters.

And how would the network itself be commanded and controlled? Any central authority, any network central citadel would be an obvious and immediate target for an enemy missile. The center of the network would be the very first place to go. RAND mulled over this grim puzzle in deep military secrecy, and arrived at a daring solution. The RAND proposal (the brainchild of RAND staffer Paul Baran) was made public in 1964. In the first place, the network would have no central authority. Furthermore, it would be designed from the beginning to operate while in tatters.

The principles were simple. The network itself would be assumed to be unreliable at all times. It would be designed from the get-go to transcend its own unreliability. All the nodes in the network would be equal in status to all other nodes, each node with its own authority to originate, pass, and receive messages. The messages themselves would be divided into packets, each packet separately addressed. Each packet would begin at some specified source node, and end at some other specified destination node. Each packet would wind its way through the network on an individual basis.

The particular route that the packet took would be unimportant. Only final results would count. Basically, the packet would be tossed like a hot potato from node to node, more or less in the direction of its destination, until it ended up in the proper place. If big pieces of the network had been blown away, that simply wouldn't matter; the packets would still stay airborne, lateralled wildly across the field by whatever nodes happened to survive. This rather haphazard delivery system might be "inefficient" in the usual sense (especially compared to, say, the telephone system)—but it would be extremely rugged.

During the '60s, this intriguing concept of a decentralized, blastproof, packet-switching network was kicked around by RAND, MIT and UCLA. The National Physical Laboratory in Great Britain set up the first test network on these principles in 1968. Shortly afterward, the Pentagon's Advanced Research Projects Agency decided to fund a larger, more ambitious project in the USA. The nodes of the network were to be high-speed supercomputers (or what passed for supercomputers at the time). These were rare and valuable machines which were in real need of good solid networking, for the sake of national research-and-development projects.

In fall 1969, the first such node was installed in UCLA. By December 1969, there were four nodes on the infant network, which was named ARPANET, after its

Pentagon sponsor. The four computers could transfer data on dedicated high-speed transmission lines. They could even be programmed remotely from the other nodes. Thanks to ARPANET, scientists and researchers could share one another's computer facilities by long-distance. This was a very handy service, for computer-time was precious in the early '70s. In 1971 there were fifteen nodes in ARPANET; by 1972, thirty-seven nodes. And it was good.

By the second year of operation, however, an odd fact became clear. ARPANET's users had wrapped the computer-sharing network into a dedicated, high-speed, federally subsidized electronic post-office. The main traffic on ARPANET was not long-distance computing. Instead, it was news and personal messages. Researchers were using ARPANET to collaborate on projects, to trade notes on work, and eventually, to downright gossip and schmooze. People had their own personal user accounts on the ARPANET computers, and their own personal addresses for electronic mail. Not only were they using ARPANET for person-to-person communication, but they were very enthusiastic about this particular service—far more enthusiastic than they were about long-distance computation.

It wasn't long before the invention of the mailing-list, an ARPANET broadcasting technique in which an identical message could be sent automatically to large numbers of network subscribers. Interestingly, one of the first really big mailing-lists was "SF-LOVERS," for science fiction fans. Discussing science fiction on the network was not work-related and was frowned upon by many ARPANET computer administrators, but this didn't stop it from happening.

Throughout the '70s, ARPA's network grew. Its decentralized structure made expansion easy. Unlike standard corporate computer networks, the ARPA network could accommodate many different kinds of machine. As long as individual machines could speak the packet-switching lingua franca of the new, anarchic network, their brand-names, and their content, and even their ownership, were irrelevant.

The ARPA's original standard for communication was known as NCP, "Network Control Protocol," but as time passed and the technique advanced, NCP was superceded by a higher-level, more sophisticated standard known as TCP/IP. TCP, or "Transmission Control Protocol," converts messages into streams of packets at the source, then reassembles them back into messages at the destination. IP, or "Internet Protocol," handles the addressing, seeing to it that packets are routed across multiple nodes and even across multiple networks with multiple standards—not only ARPA's pioneering NCP standard but others like Ethernet, FDDI, and X.25.

As early as 1977, TCP/IP was being used by other networks to link to ARPANET. ARPANET itself remained fairly tightly controlled, at least until 1983, when its military segment broke off and became MILNET. But TCP/IP linked them all. And ARPANET itself, though it was growing, became a smaller and smaller neighborhood amid the vastly growing galaxy of other linked machines.

As the '70s and '80s advanced, many very different social groups found themselves in possession of powerful computers. It was fairly easy to link these computers to the growing network-of-networks. As the use of TCP/IP became more common, entire other networks fell into the digital embrace of the Internet, and messily adhered. Since the software called TCP/IP was public-domain, and the basic technology was decentralized and rather anarchic by its very nature, it was difficult nobody wanted to stop them from joining this branching complex of networks, which came to be known as the "Internet."

Connecting to the Internet cost the taxpayer little or nothing since each node was independent, and had to handle its own financing and its own technical requirements. The more, the merrier. Like the phone network, the computer network became steadily more valuable as it embraced larger and larger territories of people and resources.

A fax machine is only valuable if everybody else has a fax machine. Until they do, a fax machine is just a curiosity. ARPANET too, was a curiosity for a while. Then computer-networking became an utter necessity.

In 1984 the National Science Foundation got into the act through its Office of Advanced Scientific Computing. The new NSFNET set a blistering pace for technical advancement, linking newer faster, shinier supercomputers, through thicker, faster links, upgraded and expanded, again and again, in 1986, 1988, 1990. And other government agencies leapt in: NASA, the National Institutes of Health, the Department of Energy, each of them maintaining a digital satrapy in the Internet confederation. . . .

ARPANET itself formally expired in 1989, a happy victim of its own overwhelming success. Its users scarcely noticed, for ARPANET's functions not only continued but steadily improved. The use of TCP/IP standards for computer networking is now global. In 1971, a mere twenty-one years ago, there were only four nodes in the ARPANET network. Today [1993] there are tens of thousands of nodes in the Internet, scattered over forty-two countries, with more coming on-line every day. Three million, possibly four million people use this gigantic mother-of-all-computer-networks. . . .

Why do people want to be "on the Internet"? One of the main reasons is simple freedom. The Internet is a rare example of a true, modern, functional anarchy. There is no "Internet Inc." There are no official censors, no bosses, no board of directors, no stockholders. In principle, any node can speak as a peer to any other node, as long as it obeys the rules of the TCP/IP protocols, which are strictly technical, not social or political. . . .

. . . ARPA's network, designed to assure control of a ravaged society after a nuclear holocaust, has been superceded by its mutant child the Internet, which is thoroughly out of control, and spreading exponentially through the post-Cold War electronic global village. The spread of the Internet in the '90s resembles the spread of personal computing in the 1970s, through it is even faster and perhaps more important.

FURTHER READING

Benedikt, Michael. *Cyberspace: First Steps* (Cambridge, MA: MIT Press, 1992).

Flamm, Kenneth. *Creating the Computer* (Washington, DC: Brookings Institution, 1988).

Forester, Tom, ed. *Computers in the Human Context: Information Technology, Productivity, and People* (Cambridge: MA: MIT Press, 1989).

Garson, Barbara. *The Electronic Sweatshop* (New York: Simon & Schuster, 1988).

Kidder, Tracy. *The Soul of a New Machine* (Boston: Little, Brown, 1981).

Long, Franklin, and Alexandra Oleson, eds. *Appropriate Technology and Social Values—A Critical Appraisal* (Cambridge, MA: American Academy of Arts & Sciences, 1980).

Penley, Constance, and Andrew Ross, eds. *Technoculture* (Minneapolis: University of Minneapolis Press, 1991).

Shaiken, Harley. *Work Transformed: Automation and Labor in the Computer Age* (New York: Holt, Rinehart, and Winston, 1985).

Star, Susan Leigh, ed. *The Cultures of Computing* (Oxford, England: Blackwell Publishing, 1995).

Sterling, Bruce. *The Hacker Crackdown: Law and Order on the Electronic Frontier* (New York: Bantam Books, 1992).

Turkle, Sherry. *Life on the Screen* (New York: Simon and Schuster, 1995).

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