IEEE

Annals

of the History of Computing

Volume 37 Number 1 January–March 2015 www.computer.org

Red Clones: Personal Computing in the Soviet Era

IEEE computer society

IEEE

Contents | Zoom in | Zoom out | For navigation instructions please click here | Search Issue | Next Page

IEEE Annals of the History of Computing

Contents

www.computer.org/annals

Vol. 37, No. 1
January–March 2015

For more information on computing topics, visit the Computer Society Digital Library at www.computer.org/csdl.

Sergey Popov operating a Micro-80 computer at the Moscow Institute of Electronic Engineering in 1979. (Photograph courtesy of Sergey Popov.)

Published by the IEEE Computer Society

ISSN 1058-6180

IEEE
Annals
of the History of Computing

Departments

Submissions
See Author Resources at
www.computer.org/annals/author.htm

IEEE Annals of the History of Computing (ISSN 1058-6180) is published quarterly by the IEEE Computer Society, 10662 Los Vaqueros Circle, P.O. Box 3014, Los Alamitos, CA 90720-1314, USA; phone +1 714 821 8380. IEEE Computer Society Headquarters, 1730 Massachusetts Ave. NW, Washington, D.C. 20036-4928, USA; IEEE Headquarters, 345 East 47th St., New York, NY 10017, USA. Subscribe to *IEEE Annals* by visiting www.computer.org/annals.

Postmaster. Send undelivered copies to IEEE Annals of the History of Computing, IEEE, Membership Processing Dept., 445 Hoes Lane, Piscataway, NJ 08855, USA. Periodicals postage paid at New York, NY, and at additional mailing offices. Canadian GST Registration No. 125634188. Canada Post Publications Mail Agreement Number 0487880. Return undeliverable Canadian addresses to P.O. Box 122, Niagara Falls, ON L2E 6S8. Printed in the United States.

From the Editor's Desk

Most histories of the personal computer have traditionally focused on the United States (and to a much lesser degree Western Europe), but as Zbigniew Stachniak reminds us in his cover article "Red Clones: Soviet Computer Hobby Movement of the 1980s," the personal computer attracted the attention of enthusiasts around the globe. In the Soviet Union, electronics hobbyists discovered the computer later than in the West, but once they did they embraced it with enthusiasm. Fortunately for them, the Soviet government approved of, and in fact supported, microcomputer-oriented groups and publications, and by the early 1980s, there was a thriving computer hobbyist community in the USSR. In tracing the shift from indigenous, Soviet-built machines such as the Radio-86RK to the mass appeal of ZX Spectrum clones, Stachniak provides an important contribution to the growing literature on the global history of the personal computer.

In his overview of the history of computing in India in the period 1955—2010, Vaidyeswaran Rajaraman extends our global perspective in a different direction. His careful attention to the larger political economic context of technological innovation reminds us that, as first a newly independent nation and later as a developing economy, India necessarily pursued a different strategy from those of the Western industrial powers. The long-term investments that India made in education proved particularly fruitful, and they eventually allowed India to assume its current status as a powerhouse of software development and services.

Speaking of educational infrastructure, in his insightful and very personal history of the IBM Sales School, James Cortada provides a vivid reminder of the central role of the distinctive "IBM culture" in the success of that organization. This culture has often been mocked by early personal computer entrepreneurs as stale and old-fashioned, but Cortada reveals a lively, productive, adaptive, and extraordinarily durable system that served to capture, disseminate, and create new knowledge within the firm. "There is no saturation point in education," said the sign above the IBM Education Center in Endicott, New York, and although the wording might today sound antiquated, the sentiment would be familiar to any modern start-up company.

In our third article dealing with education (the emerging theme was unplanned but auspicious), Peggy Kidwell describes how technical workers learned to use slide rules, as well as how slide rules were used to learn. Although the Scottish mathematician John Napier first published his work on logarithms in 1614, and instrument makers were producing slide rules within a few decades after that, it was not until the late 19th century that slides rules became an important part of American mathematical practices. As with all of the history of computing, the technology alone was not enough to drive history: equally important were schools, textbooks, educators, and manufacturers.

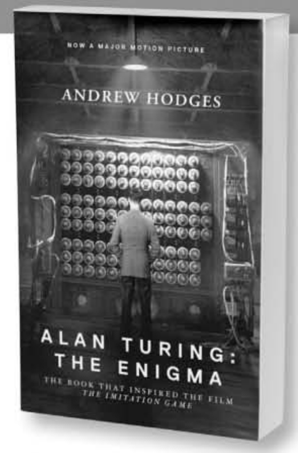
The sketch map of the early ARPANET that first appeared in 1969 has assumed iconic status among historians and the public alike. It features in many a book, website, documentary, and museum exhibit illustrating the "origins of the Internet." In their article on the production and interpretation of this and other ARPANET maps, Bradley Fidler and Morgan Currie trace the assumptions, strategies, and meanings associated with such representations. Like all maps, the ARPANET maps are historically situated documents; they cannot be seen as merely descriptive, but are intentionally designed to highlight and reinforce certain values. In the case of the 1969 ARPANET map, for example, the visual representation emphasized decentralization and equality among nodes, and it concealed hierarchies of login access and the directionality of communications. There is a growing literature on the material culture of information, software, and other virtual artifacts, and Fidler and Currie provide us with an exemplary case study in the value of such analyses.

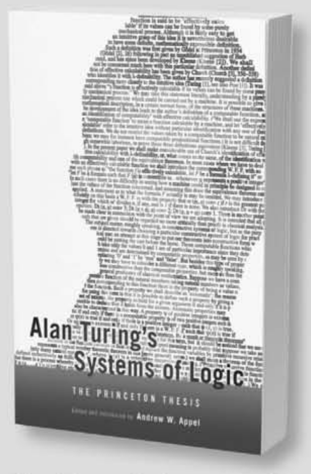
On a final note, this issue represents the first for which I am the nominal editor in chief. But as I only assumed that role on the first of this year, all of the real work was done by David Walden, who for the past several issues has served as not only the acting EIC of the *Annals*, but in a variety of other informal roles as well. I cannot thank Dave enough for his contributions to the *Annals*, and his managerial experience, personal and intellectual enthusiasm, and work ethic have set a high bar for me to reach for over the next few years. All of our authors, editors, reviewers, and editorial board members have gone above and beyond the call of duty this past year. Thanks to them, and to our tireless and amazingly competent IEEE CS staff, we have another excellent volume of top-quality articles and departments to reflect proudly upon and another year of the *Annals* to look forward to.

Nathan Ensmenger *is the editor in chief of* IEEE Annals *and an associate professor in the School of Informatics and Computing at Indiana University. Contact him at nensmeng@indiana.edu.*

 Published by the IEEE Computer Society

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®


NOW A MAJOR MOTION PICTURE
ANDREW HODGES
ALAN TURING: THE ENIGMA
THE BOOK THAT INSPIRED THE FILM THE IMITATION GAME


Alan Turing's Systems of Logic
THE PRINCETON THESIS

Gordon Bell

Dag Spicer
Computer History Museum

Editor: Dag Spicer

Computer pioneer Gordon Bell has been one of the industry's leading figures for nearly 50 years. A fellow of the American Academy of Arts and Sciences (1994), American Association for the Advancement of Science (1983), ACM (1994), IEEE (1974), and Australian Academy of Technological Sciences and Engineering (2009) and a member of the National Academy of Engineering (1977) and National Academy of Science (2007), Bell was also a founding board member of the Computer History Museum.

The following interview is based on an oral history conducted by Gardner Hendrie for the CHM in June 2005. (The full transcript is available in the CHM archive.[1])

***Gardner Hendrie*: Could you tell us a little bit about where you were born and your family and give a little background about your formative years?**

Gordon Bell: I'm a fan of crediting everything to my parents and the environment that I grew up in. I was born in Kirksville, Missouri, on 19 August 1934. Kirksville was a college town and a farming community, with a population of about 10,000, and it hosted the Northeast Missouri State Teacher's College, which eventually morphed itself into Truman State University. My father had an electrical contracting business and appliance store and did small appliance repair. I grew up working at "the shop" and spent my formative years in that environment.

***Hendrie*: What are the earliest memories you have of thinking about what you might want to do when you grew up?**

Bell: I was one of the best electricians and appliance (e.g., dishwasher) repair persons in the town when I went to college. The Rural Electrification Administration (REA) program to electrify all the farms in the country had been established in the mid-1940s, so I did lots of house and building wiring. I installed industrial equipment and worked on all of that kind of stuff, such as appliances, buildings, houses, and industrial machinery.

***Hendrie*: Tell me about the things you did, what you studied. You obviously learned a lot from just working with your father. What did you study in high school?**

Bell: What was really important was having a wonderful science teacher and a wonderful math teacher. I still remember both very fondly. At that point in time in Kirksville, Missouri, you didn't take calculus since it wasn't offered, but I took chemistry and physics and then geometry, trig, and (perhaps) solid geometry. Those were really critical to enable me to pass the SAT and go to MIT. At some point, maybe when I was 12 or so, I thought I wanted to be an engineer. I had no idea what an engineer was. I had books that I sort of read—*Books of Knowledge* and *The Way Things Work*—so I gleaned that somewhere, somebody figured out how to make these things work and that was the interesting thing, not repair. Repairing them was okay, but in fact, designing them or inventing them seemed like a lot more fun. So that was basically the course that I set fairly early, with no one telling me I should be doing this.

I really had no trouble at MIT even though one of my dad's golf buddies who taught at the college advised me not to go there because I might fail. MIT was hard work, and I have nice memories about being there even though I can't imagine being admitted now. I went into the co-op program because I wanted to understand what it was like to be an engineer.

***Hendrie*: So when you were approaching graduation, you must have been thinking about where you were going to go and what you were going to do. Did you ever think you wanted to continue an academic career, or did you want to go out and get a job?**

Bell: The problem was the co-op experience had convinced me that I didn't really know if I wanted to have a job living in a sea of desks with other engineers, so this is where serendipity kicked in. A really good friend and graduate year roommate, Bob Brigham (for whom my son, Brigham, is named) and I walked into the department head Gordon Brown's office. He said, "Well, what are you going to do with your lives at this point" or something like that. He went on and offered: "Why don't you guys go to Australia and help my friend, Rex Vowells, who's starting a computer engineering program in their EE department? They've just got a computer at the University of New South Wales. It's an eight-year old university and wants to pattern itself after MIT. Go there and do some research, teach some courses."

The Fulbright program accepted us and it was a wonderful experience. When I visited the University of New South Wales a few years ago to give a lecture, outside of the department head's office they had a keypunch and a big reproducing card reader/cardpunch. I said, "Gee,

 Published by the IEEE Computer Society 1058-6180/15/$31.00

did that come from the DEUCE [Digital Electronic Universal Computing Engine]?" And they said, "What's the DEUCE?" I explained, "That was the English Electric DEUCE, named the University of Technology Electronic Computer (UTECOM). It was about the second or third computer brought into Australia. I programmed it using that card equipment."

Anyway, Bob and I spent a year there programming DEUCE; Turing had worked on ACE at the National Physical Laboratory that English Electric used as a prototype for DEUCE.

Although we had an enjoyable time seeing the country and going to the beach, we worked very hard. Bob and I wrote a program we called the Symbolic Optimum DEUCE Assembly (SODA). DEUCE was a very difficult machine to program because it had a working store that held the program of eight, 32 delay lines, backed by an 8,192 word drum. The philosophy of Turing was don't waste any hardware on what people can do. Make the people work, not the hardware. And so the instructions coded the opening of gates for delay lines to go into other delay lines. It was like you wrote this little traffic program to move a word at a given point in time from one delay line in through a gated adder and into a delay line register.

Australia is also where I met my wife, Gwen, another Fulbright scholar working on city planning. I proposed to her with the DEUCE. I wrote a program that was essentially just a little flowchart. The way the display worked was that you could bring down stuff into one of the memories. DEUCE had a 32 × 32 scrolling display grid that you could write messages into it. Thus, the memory was also an output and user interface.

***Hendrie*: You got her in front of the machine?**

Bell: Yes, I said, "Here, run this program." Since the machine had no keyboard, all she had to do was answer yes or no questions by flipping a couple of sense switches. I even think I may have submitted the program to the library.

***Hendrie*: Did you know what you were going to do when you came back to the United States?**

Bell: I left Australia in August 1958. I don't think I knew exactly what I was going to do. Gwen had to finish her master's degree at Harvard, so location was constrained to Cambridge.

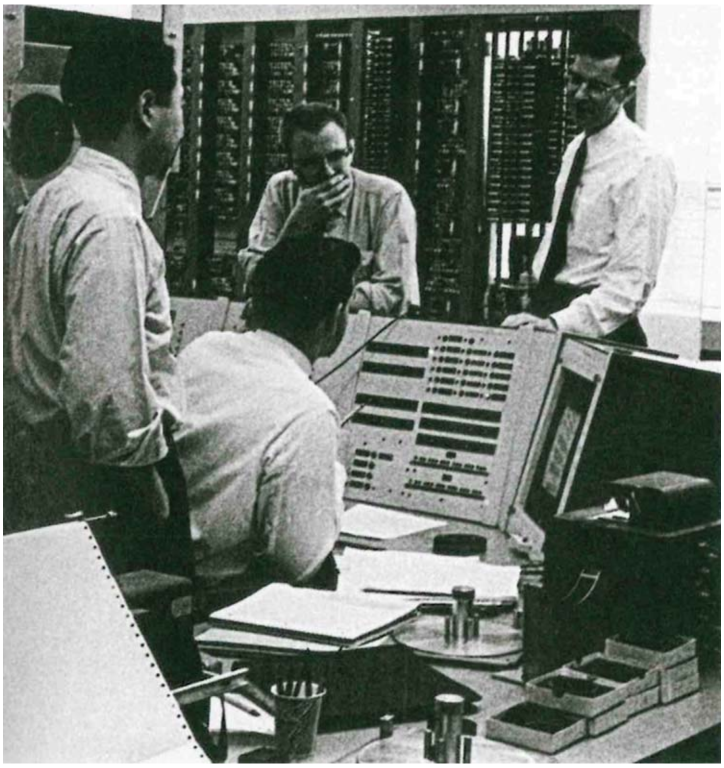


Figure 1. Gordon Bell and colleagues at a TX-0, circa 1959.

I interviewed at MIT, and I recall interviewing at EPSCO, which made test equipment. And I had several other offers. Philco in Philadelphia was interesting as an early computer manufacturer. I don't recall an offer, but GE was a possibility because I had been a co-op student there. I had an offer from NCR after a written half-day intelligence test. At some point, I interviewed at BBN with Ed Fredkin and JCR Licklider (Lick), but luckily they didn't hire me to do programming research.

So, my first real job was as a staff engineer at MIT working for Professor Ken Stevens on speech analysis where I wrote a program analysis called analysis-by-synthesis that is still referenced. I also worked at MIT's Instrumentation Lab on a pulsed analog correlator—an interesting, but flawed idea.

***Hendrie*: The DEC PDP-1 had not been built yet, had it?**

Bell: No, no.

***Hendrie*: So, MIT Professor Jack Dennis didn't have his PDP-1 yet?**

Bell: Right, in 1960 a tape controller was needed for the TX-0 (see Figure 1) and that's

> **DEC's PDP-6 was a response to DEC's cofounder Harlan Anderson telling Bell, "Gordon, go off and look at building a serious machine."**

how I got to Digital Equipment Corporation (DEC). I was buying modules from DEC for the tape controller design, and that's how I met the company and ended up getting a job offer in May 1960. I interviewed with Ben Gurley, the PDP-1 designer who had critiqued my tape controller design.

Dick Best was their chief engineer, doing circuits, and then in that process I met Ken [Olsen] and Andy [Harlan Anderson]. I was the second computer engineer working for Ben and had badge number 80. DEC didn't have the problems that I had seen as a co-op student at GE where I had to live in a sea of desks. Everybody had their own little office made from hollow core doors, which was uncommon.

I know I wrote a floating-point package for the PDP-1 and was instrumental in starting DECUS in 1961. I knew we have to share programs among the users—just like IBM's Share. There's so much code to write.

Fortunately, DEC got a big order from International Telephone and Telegraph (ITT) that allowed us to continue and grow. It would have been 1961, and we made a deal with ITT to build them a switching system called the ADX 7300 to replace their "torn Teletype" tape-switching systems. As a project engineer, I learned about serial communication and ended up inventing the UART (Universal Asynchronous Receiver-Transmitter).

Half of the PDP-1s were sold to IT&T under the ADX label. And in fact, it was one of the key things that I'd say "saved the company" and made it a computer company because just getting a lot of "standard" orders was the key thing. Otherwise, we could have just been special systems builders.

That's also about the time you and I met, and I started working on the PDP-4 [minicomputer]. And I think you [interviewer Gardner Hendrie, who worked at Foxboro, an industrial controls company] triggered the PDP-4, which was to say, "We want a control device." And we said, "Yeah, we can make one of those," and I became the project engineer of the PDP-4.

For whatever reason, I decided that it wasn't going to be PDP-1 compatible. This was before I understood about programming and programming cost and investment—still probably the most misunderstood concept in computing. People just don't understand the value of software, the integrating effect of software, and why you, whenever possible, use somebody else's program and interfaces where there's an installed base of software that you have access to. Changing the architecture is always tempting for whatever reason, and it's usually a bad idea.

One of the quotes that [Maurice] Wilkes made after completing the EDSAC was, "It wasn't very long before we began to realize the value of the software that we had created would vastly outweigh the cost of the machine." Anyway, the PDP-4 turned out to be the resulting architecture that was used for the PDP-7, the PDP-9, and the PDP-15 implementations. It had a long, two decade life.

***Hendrie*: When did the first ideas for PDP-6 come about?**

Bell: I'm sure Andy [Harlan Anderson, DEC's cofounder] triggered it and said something like, "Gordon, go off and look at building a serious machine," and that this should be a bigger machine, a machine designed for time-sharing. We'd been playing with time-sharing with BBN, and the MIT CTSS was running. The PDP-1 was way too small to be worth time-sharing. There wasn't enough to share. It didn't have floating point and with an 18-bit word, little computation ability. So the idea was to improve on all of those dimensions, a lot more bandwidth, a lot more expandability, a lot bigger machine, and make a real computer that could handle numbers. That would have been in 1963.

The time-sharing idea was floating around then. [John] McCarthy and [Marvin] Minsky had described time-sharing at MIT, and then Corby [Fernando Corbató] had built CTSS using the 7090. So the ideas at MIT and then the PDP-1 time-sharing at BBN were the early, very first time-sharing machines, and then Dartmouth built their Basic using the GE

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

computers. Those were the ideas. PDP-6 was the first time-sharing computer that was designed from the bottom up (see Figure 2).

Alan Kotok and I started the project, and as it picked up steam, two circuits engineers and technicians joined. There was the idea of having general-purpose registers. That's kind of the big thing that we had architecturally. We didn't know it at the time, but the Pegasus in the UK had a somewhat similar kind of architecture and then the IBM System/360 used general registers.

***Hendrie*: Was there the concept of a protected mode or anything like that?**

Bell: Oh, yeah. This was the big part of the architecture. We knew time-sharing needed to have the protection among users as well as some way to share common code. This ability was added in the PDP-10 with the addition of another set of relocation registers. Ultimately, the machine evolved to have a page table.

When the IBM System/360 came out in 1964, we were lucky that their design wasn't oriented to time-sharing. We delivered 10 PDP-6s by June 1966 (1,000 of the family were built). One of the first deliveries was to Stanford. Of course, the first thing John McCarthy said was "I want a large, fast memory. I am going to go out and buy one. DEC is charging a zillion dollars for memory. Ampex, will you design us a large, inexpensive core memory?" So we probably started the add-on memory business. The PDP-6 was easy to interface to, and the specs were printed right in the programming manual.

***Hendrie*: Did you get any ideas from the Atlas?**

Bell: Yes. The Atlas to me was just a spectacular machine. I was really impressed by it. I remember visiting Manchester in 1962 or so, and then I saw it being built at Ferranti, in a very casual way. Later I went to Manchester and I was watching these guys build it. I sort of said, "Gee, if these guys can build a computer like that, I mean, this is crazy, you know, we can build anything we want. We should be able to do this." Because they were just very casual. I asked, "When are you going to get this thing wired?" They weren't in a hurry or anything like I expected from a commercial company. "Well, you know, it'll maybe be six months before we can turn the power on."

Almost from the start, hand-wiring of machines was a limiter for production. It was especially true for building PDP-6s that had lots of interconnections. Either you have to have a machine to check the wiring, or you wire it automatically. And so we called up Gardner-Denver and got them in and said, "You know, we'd like to buy a machine." That was the beginning of the PDP-8 minicomputer. Because the main thing about the PDP-8 was that it was mass produced using wire-wrap. That all came about because we couldn't manually wire PDP-6s.

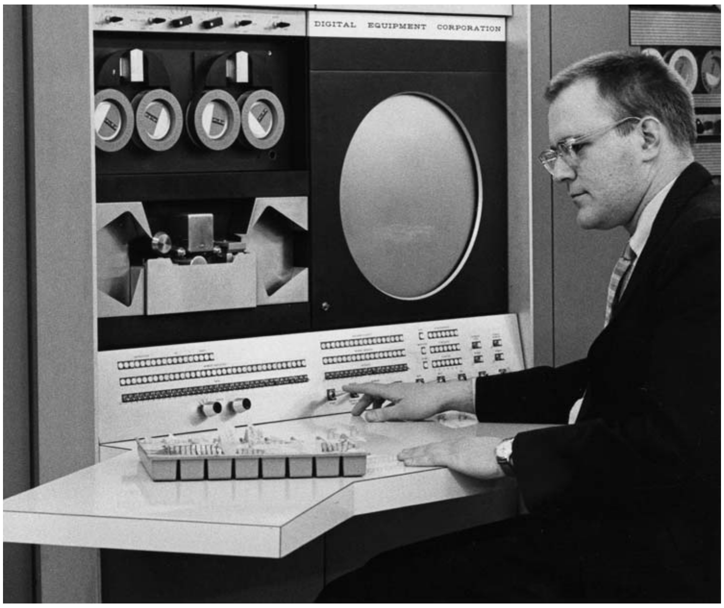


Figure 2. Gordon Bell operating a PDP-6, circa 1964.

***Hendrie*: What did you do next?**

Bell: I think at that point I didn't feel the machines were challenging, and I saw what DEC had to do, which was to make copies of their existing line of computers. At this point, I had discovered company growth would now be a software problem and that you've got to take advantage of that. DEC should not be building any new architectures. It shouldn't be looking at architecture as a way to solve anything. That prompted me to take a sabbatical. I wanted to go back to academia, and Ivan Sutherland and I talked about doing something together. I wasn't that interested in graphics, and I think he was going to Utah, so he suggested I go to Carnegie Tech because he had been a grad. So I went to meet Perlis or Newell.

***Hendrie*: Were Newell and Simon both there?**

Bell: Allen Newell, Herb Simon, and Alan Perlis were there. The three of them had written

PDP-6 was the first time-sharing computer that was designed from the bottom up.

a paper the summer I got there titled"What Is Computer Science," which to me is a classic. It was a little paper for *Science* about trying to define the scope of it and why it should be called computer science. I went for an interview there, and they offered me a job as associate professor. Rod Williams was the head of EE, where I had a joint appointment, and Perlis was heading the Computer Science Department. I convinced Allen Newell to help me write the book *Computer Structures*—that was my main work. During this time (1966–1972), I consulted with DEC.

Writing the book *Computer Structures* and working with Allen was a great experience. We invented notations for describing computer systems and constructed a taxonomy that formed the basis of The Computer Museum collection that was started when I returned to DEC. The book helped with the inventions of the PDP-11 Unibus and general registers concept. These got invented by just thinking about them for the book. In 1971, Harold McFarland, an undergraduate student who I worked with, went to DEC and took the basic concepts to architect the PDP-11.

In 1972 I had it arranged to go to Australia for a sabbatical next year. At that point, I believe Win Hindle of DEC made me an offer: "Why don't you just come back and run engineering? We just need somebody to take care of all the engineering." I basically thought, "I'm emotionally packed. I might as well go back to Maynard." However, the main motivation was that the microprocessor had just been introduced and VLSI was on the horizon as a technology that would yield spectacular results.

Ken initially assigned me two engineering groups—power supplies and memories—because those were the parts that were centralized or that could be centralized. Also, I knew nothing about them. It made me humble. By 1974 I was VP of all R&D.

***Hendrie*: Tell us about the VAX. Who was on this original team, who were the key people?**

Bell: VAX stated in April 1975 with the VAX-A team of Dave Cutler, Richie Lary, Steve Rothman, Bill Strecker, and me. Tom Hastings joined as the scribe to keep the record. Bill had been in research and had outlined a proposal for a byte-oriented architecture we called "culturally compatible" with the PDP-11. I named it the VAX-11 to keep us on track in terms of it being an extension to the PDP-11—of the same style but with more bits, but you'd still recognize it as a PDP-11. The other decision we made was to put a PDP-11 in the VAX so there was a mode that executed PDP-11 code in a part of memory. That was essential because we ran a lot of software as a PDP-11 for a long time.

We didn't see the switchover to a cache-based RISC-type machine happening within decade. In fact, we were going to do the opposite, put all of that code in RAM, because at the time it was just peaking where you could basically put anything you wanted into microcode. It was the opposite of RISC. It was the most complex machine you could build. It was used for both business and science, so it was both for Fortran and Cobol. It had Cobol instructions, just like the 360 and Burroughs B5500. This is one of the things that most academic architects never bothered to understand. The VAX-11/780 hardware team had built a general-purpose microcode emulator. By September 1975, we had a running machine—a fast, or relatively fast, running machine that we could check the microcode on and test programs.

The IBM System/360 had a number of operating systems depending on how big your company was, and what I wanted was a single operating system in which the only thing that was different was what you might run on that and how many people were attached to it, down to single user.

By the summer of 1978, after a year of sales, it was clear that the machine was going to be successful, so at that point I proposed and wrote the "VAX Strategy Memo," which essentially bet it all on VAX. We would build a number of machines and would stop building PDP-1s or 8s except to maintain designs as long as they were profitable. The strategy didn't say anything about the PDP-10. We stopped building PDP-10s later, and everybody accused me of killing the PDP-10, but in reality what happened was that there was no one able to build PDP-10s. Alan Kotok had stopped working on PDP-10s, and now no one knew how to build a PDP-10.

The VAX Strategy was the big event in the fall of 1978 that came while I was diving on a sailboat in Tahiti. It was really an "Ah ha" moment. I thought, "We've got to build a range of machines but unlike the IBM System/360." There were two things that were different. In the case of the 360, all the models were fundamentally mainframes. They were all operated the same way. They had different operating systems. In the case of VAX, they all had the same operating system, they had the same environment, and the thing that differentiated them was where they lived in the hierarchy. So we pioneered clusters by aggregating PDP-11/780s and then bigger machines together saying we would never be able to afford building the big machines that IBM was building and that the way to solve that problem was simply to cluster things. That lesson got forgotten later on and that was one of the bad parts of what happened to DEC.

The irony is that all post 2005 computers are built as clusters whether for the cloud or for supercomputing.

***Hendrie*: Were there any issues in the software development of the original VAX?**

Bell: Software went very smoothly with Dave Cutler leading VMS. Dave had written RSX-11 and, after leaving Digital, led Microsoft's NT. There weren't research questions to answer. Bell Labs had gotten an early VAX, and I think that they had ported a Unix System 3 or 5, whatever, to it. Then we had VAXen that went to Berkeley for Berkeley Unix.

***Hendrie*: Tell us about DEC's efforts at building a personal computer.**

Bell: DEC introduced three separate personal computers in 1982 when the world was clearly just standardizing on the IBM PC. This was really the root of the problem, and you didn't have to be very good either because everybody had adopted the standard. So I'm not going to fault us for doing the three systems because there was, for example, the DECmate, that was based on the PDP-8, and which was a fairly simple thing because of the tiny engineering group it had. They were very productive. They had customers, they had legacy, and they had software, so we weren't investing a lot. It was used both for small business and word processing. In fact, I tend to look at it as the DECmate was a wonderful word processor, and it competed with Wang. It was very nice, so I wouldn't have changed that.

The PDP-11 was our bet on what was going to be the workstation PC, and in fact, as it came out it was superior to the PC—there's no question about that—but the whole issue of standards really overwrote everything else. The other thing was that we didn't even allow people to connect to it. The PDP-11-based workstation, called the PRO, was held proprietary so there was no way to compete with an open standard like the PC where every part was open.

The Intel-Microsoft PC was the Rainbow. I don't know how long it was before it was clear that that's the way the industry's going to structure, but it wasn't very long after the PC—clearly by 1983. And DEC never quite got it. It took years. I know they didn't get it by 1986. I was at NSF and Ken [Olsen] sent a PC to me to test. It had everything but software, and I said, "Where's the software?" Ken said, "We need special software because it doesn't run IBM PC compatible software, or even the plain Microsoft DOS." DEC was not used to following standards it didn't create. DEC might have been a player in the PC like HP or anybody else that was starting up because it was a volume issue and DEC was okay in manufacturing. DEC could get itself together to manufacture when it had to, and it would have been a great challenge, but when they finally did do it they were always behind. I'd say certainly the PDP-11 workstation was a waste and it didn't have the right stuff, but in this case it was really the PC coming in as proprietary, when what the world wanted was standards.

Worst yet was the workstation market being developed by Apollo and Sun. These took critical parts of the VAX market.

***Hendrie*: Let's get back to your career.**

Bell: I left DEC after my heart attack in the summer of 1983. I'd say at that point, that summer or shortly thereafter, it was pretty clear to me that the PC was the standard. I came back from the hospital, and I sat in on some very contentious meeting about something and I thought, "You know, this is just too hard." This will kill me.

***Hendrie*: From your own personal point of view, you decided that you probably didn't want to stay around DEC and decided to leave. What did you do?**

Bell: Henry Burkhardt came to me and said, "Hey, I'm starting this company with Ken

Between 1983 and 2007, I invested time and/or money in over 100 companies or about four companies per year.

Fisher. Do you want to join us as a cofounder?" Basically, I talked with the guys and it was an insane business plan—about as far from the "ideal" I eventually wrote about in the book *High Tech Ventures*. But my attitude was, "Hey, I'm leaving, here's a train. I'll just get on this train, and if I don't like the train, I'll get off. What the hell. No big deal. Here's a start-up. Let me see what that's all about." So we formed Encore. It didn't take me very long to get educated about start-ups, having never been part of one since the founding of DEC. By February 1986, we had a set of products including a multiprocessor mini, a workstation, and networking infrastructure that were quite impressive. However, the board fired Henry, so I left the company. Sun eventually acquired Encore for patent rights.

In December 1985, I decided to move to the West Coast. I had been there as a board member of Silicon Compilers and met Steve Blank, Ben Webreit, and Allen Michaels, who had just sold Convergent, a successful start-up they had formed, and suggested using the MIPS chip to build a powerful workstation. The company Ardent built a great graphics supercomputer based on the Cray vector architecture. In the process, I gained an even greater respect for Seymour Cray. We sold several hundred, but in the end, SGI had more end-user software and was simply too entrenched.

Meanwhile in early 1986, Erich Bloch had recruited me to start up the National Science Foundation's Computing Directorate. Various computer science areas from NSF were combined to form the Computing and Information Science and Engineering (CISE) Directorate. By 2008, the budget was maybe a half a billion dollars for computing research and one of the largest directorates. It included hardware and software engineering and computer science research plus supercomputing centers that were just forming. Each supercomputers center funded its own network. That was really a fun time. There were alternate days I felt like somebody competent could actually run the country, although it was nontrivial, but I learned a bit about how things worked in Washington and what the executive branch did. I still get a queasy feeling when I go to Washington.

One of the first and maybe most important actions was to create a network division that worked on the National Research and Education Network (NREN) Plan (aka the Internet). That plan was used to get funding for regional networks and to tie these through a funded backbone. This was in response to the Gore Bill [Supercomputer Network Study Act of 1986]. NSF's and Al Gore's staff wrote a bill that said that NSF should write a plan for a national networking infrastructure. I led the cross-agency group from Department of Energy, Department of Defense, National Institutes of Health, NASA, and NIST that created the plan for the Internet that was actually executed.

I think what CISE did, in addition to coalescing the funding of computer science research, was establish networking and get the supercomputing program on the right foot, including a painful switch to a more standardized environment. One of the more controversial goals for CISE was a focus on parallelism. Perhaps the NREN response to the Gore Bill was most important because it established the Internet.

A by-product, resulting from the discussions about parallelism at NSF, was the establishment of the Gordon Bell Prize for parallelism in 1987. The prize is administered by the ACM at part of the annual ACM/IEEE Supercomputing Conference where the results are presented. In 1987, the first prize was for about one-half gigaflop and parallelism of a thousand. In 2014, performance is about 50 petaflops and parallelism of several million. I believe the prize is a useful part of supercomputing and personally very gratifying.

After NSF, I began enjoying life and just floating around Silicon Valley. I was on Suhas Patil's board at Cirrus Logic and doing other things including consulting at Sun and Intel. The net result of this period, 1983 to 2007, was I invested time and/or money in over 100 companies or about four companies per year. A nice result of this period was the creation of the Bell-Mason Model for new ventures with Heidi Mason (described in *High*

Tech Ventures). Heidi is the CEO of the Bell-Mason Group that helps companies manage and measure their ventures.

In 1994, I met Jim Gray, one of the happiest and most productive events of my life—we had never met before. He came to my house, and we talked about what computers were going to be like and decided we had exactly the same religion about how computers were going to be built.

We were convinced that scalables were going to take over all the other computer classes. All the different little classes and niches were going to be wiped out and replaced by scalable PCs or "bricks." Clusters of them were going to wipe out workstations; network-connected PCs were going to go right up through to minis and mainframes and eventually supercomputers. Then Jim wrote that up, and he gave a talk about it in Berkeley in three McKay lectures. That was kind of the beginning of our belief.

Jim had been consulting, and he called me to say, "I can't deal with this. I just need a place to work and a project. This is just not any good for me." In 1991 I had helped Nathan Myhrvold start Microsoft Research and recruit Rick Rashid and was on their advisory board. So I emailed Microsoft to say, "Hey, we've got to hire Jim and set up a lab." It turned out he had been talking with them for a year or so, so probably all I did was to crystallize the charter. I'm not sure whether Nathan asked me to run Microsoft Research or not, but in any case, I do recall saying, "I'll help find someone really good to do it!"

It took Microsoft about a nanosecond to hire Jim in the summer of 1995, and then Jim called me to join. "Okay, you've got to be part of the lab. You need a job, you know, your life is just floating around doing whatever. You need discipline." So he convinced me to come with him to the newly established Microsoft Bay Area Research Center in San Francisco. In August 1995, I joined to work on "telepresence." Jim and I convinced Jim Gemmell to join me for the "heavy lifting."

In 1999, I decided to scan all my "bits" [personal and professional records] and that got me onto the path of "cyberization." Since then, I think I've captured all the personal bits I have into our version of Vannevar Bush's Memex, which Gemmell and I called MyLifeBits. In 2007, we ended the program with the book, *Total Recall*. Now many companies are building products that help people put various parts of their lives into cyberspace.

***Hendrie*: What are some other things you are proud of?**

Bell: As an engineer almost everything looks like a project to me: I'm proud of nearly all of them in some fashion, even the dumb ones, because they taught me something. Some are more important and lasting, but I like to work on the new ones that others don't see. That's basically how I conduct myself, as an itinerant project engineer looking for another useful and novel computing artifact to be build. No "me too," projects!

I feel bittersweet about DEC because I wanted it to at least outlive me—especially since it had achieved the number two industrial position for almost a decade. The incompetence that caused its demise used to make me angry. But the people, many machines that were delivered, and contributions to computing are something to be proud of and maybe all an engineer can ask for.

Being proud of working with great people and mentors in academe, government, and industry has been a highlight of my life. Simply finding great people who have allowed me to collaborate to build interesting artifacts, companies, or whatever is something to just feel lucky about.

The Computer History Museum is something I feel very proud of, helping start an institution for preserving the artifacts and stories about computing starting with the first exhibit that Ken Olsen and I created at DEC in 1975, followed by The Computer Museum that Gwen created in 1979. The museum goes beyond just a successful project (how I measure a professional life) because it has a chance to live forever (even as the computer disappears) as an important institution. The museum may be at the top of that list.

So for me, its three things: Computing, My Life; and Computing My Life.

Reference and Note

1. See the CHM archive, http://archive.computer-history.org/resources/text/Oral_History/Bell_Gordon_1/102702036.05.01.pdf, for the full transcript.

Dag Spicer *is the senior curator at the Computer History Museum. Contact him at spicer@computerhistory.org.*

Red Clones: The Soviet Computer Hobby Movement of the 1980s

Zbigniew Stachniak
York University, Canada

The Soviet microcomputer hobby movement began in the early 1980s. It passed through two main developmental stages before it melted away a decade later, leaving behind a country enchanted with microcomputing. This article traces the development of the movement and assesses its significance and place in the USSR's history of computing.

The political and economic barriers erected by the Cold War affected more than the flow of Western dual-use technology (that is, technology with both military and commercial applications) to the Soviet Union and other East European countries. They also delayed the dawn of personal computing in the Soviet bloc. By the end of the 1970s, there were still no signs of microcomputers on the USSR's consumer electronics market, and there were no computer hobbyists who could reshape the Soviet society's perception of computing as successfully as North American computer enthusiasts reshaped theirs. However, the barriers did not stop the flow of technical information entirely, and a decade later, the Soviet computing landscape was filled with myriad home computer brands, thanks to the dynamic computer hobby activities that sprang out of Moscow, Leningrad, and other large Soviet urban centers. The main focus of this article is on that movement—the movement that should not have happened.

The conditions that initiated the North American computer hobby movement of the 1970s (such as access to microprocessors and information about them) were mostly absent in the USSR, and that, at least partially, explains why Soviet electronics hobby activities of the time did not include building and experimenting with computers. By the end of the 1970s, the Soviet electronics industry was in a position to pick and choose among the many proven Western designs for home and personal computers; the selected designs could have been copied and mass manufactured to move Soviet society swiftly into the microcomputing era, skipping the hobby computing phase all together. Indeed, that's what had happened in East European countries such as Bulgaria. The ПРАВЕЦ 8x [Pravetz 8x] series microcomputers were Bulgarian clones of the Apple II+, IIe, and IIc computers. Mass manufactured from 1982 until 1994, they provided computing support for many sectors of the Bulgarian economy and its educational system.[1] That, it seems, prevented a large-scale computer hardware hobbyism in Bulgaria.

Things did not happen in exactly that way in the USSR, however. In the early 1980s, at the time when North American computer hobbyists were unplugging their soldering guns, Soviet electronic hobbyism suddenly branched into computing, a development that eventually had a profound effect on Soviet society's introduction to computing. Why did that happen? What was the much-delayed Soviet computer hobby movement to achieve? In this article, I address these questions by analyzing the movement's origins and developmental stages, and I attempt to assess the movements place in the USSR's history of computing.

Could Soviet Computer Hobbyism Have Emerged in the 1970s?

The subject of the North American computer hobby movement of the 1970s and its impact on the creation and shaping of the personal computer industry has received wide coverage in scholarly and popular publications and need only be recapped briefly here.[2] The movement grew out of a more than half-a-century-long tradition of radio and electrics hobbyism backed by a variety of hobby magazines such as *Modern Electrics* (renamed

 Published by the IEEE Computer Society 1058-6180/15/$31.00

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Electrical Experimenter) and *Popular Electricity in Plain English,* both launched in 1908; *Radio-Craft,* first published in 1929 and renamed *Radio-Electronics* in 1948; and *Popular Electronics*—perhaps one of the most influential hobby electronics magazines of the last century—which was launched in 1954.[3] Since the end of the 1940s, computer enthusiasts and dedicated educators had been involved in a range of computing-related activities from the design of computer toys and educational aids to publishing and setting up computer social groups and organizations.[4] The introduction of the first 8-bit microprocessors onto the market in the early 1970s triggered the outbreak of homebrew computer activities that spawned the North American computer hobby movement. Technological advancement in the semiconductor industry was as important to that process as were the strength of electronics hobbyism and the presence of the intellectual drive to redefine the social status of computing. In the words of historian Paul Ceruzzi, "When these forces met in the middle, they would bring about a revolution in personal computing."[5]

The Soviet radio and electrics hobbyism tradition also goes back a long way. The first publications for radio amateurs started to appear in the 1920s. *Радиолюбитель* [*Radio Amateur*] and *Радио всем* [*Radio for Everybody*] were launched in 1924 and 1925, respectively. In 1930, they merged into a single publication, *Радиофронт* [*Radiofront*], which in 1946 became *Радио* [*Radio*] and was arguably the most popular Soviet electronics hobby magazine.[6] During the Soviet era, radio and electronics hobby activities were well-supported by the state, which sponsored inventor clubs and national exhibits. The prestigious All-Union Exhibit of Achievements of Radio Amateurs-Constructors showcased the achievements of Soviet inventors from the mid-1930s. However, if one excludes the black market in electronic components, there wasn't much more on the Soviet electronics hobby landscape of the 1970s. In drastic contrast to space exploration themes, computing was mostly absent from Soviet science and technology posters and postage stamps, which were powerful propaganda tools during the Cold War era. Therefore, it was a dearth of information about computers and semiconductor products, plus an absence of computer educators, enthusiasts, and hackers, that painted the backdrop for the Soviet electronics hobby activities. What the hobbyists lacked the most was access to novel semiconductor devices such as microprocessors and to information about them. In the 1970s, *Radio* did publish articles on digital electronics, but it never ventured into the world of microprocessors in any significant way. For instance, one of the earliest series of educational articles explaining modern computers to electronics hobbyists, published in *Radio* in 1978, only mentioned the microprocessor as a novel integrated CPU device.[7]

In a large geopolitical context, the microprocessor's absence from experimenters' workbenches on the Soviet side of the Iron Curtain was one of the repercussions of the Cold War. Since the advent of the war, the United States and its NATO allies had tried to restrict the flow of militarily strategic technology to the Soviet Union and other member countries of the Council for Mutual Economic Assistance (or CMEA).[8] In 1949, the NATO countries (excluding Iceland) and Japan formed the Coordinating Committee for Multilateral Export Controls (CoCom), an informal nontreaty organization whose objective was to set up a comprehensive system of export controls and enforcement measures to restrict the transfer of those technologies and products to CMEA countries that could significantly advance military capabilities of the Soviet bloc. Furthermore, several countries introduced unilateral export control laws for national security, foreign policy, and economic reasons.[9]

While export controls impeded the progress of CMEA's computer industry and widened the technological gap in the area of computing, they did not entirely stop the flow of Western computer technologies and products through the Iron Curtain,[10] as exemplified by the Единая Система [Uniform System] computer development program.

In 1969, the USSR and the majority of other CMEA countries signed the multilateral agreement on collaboration in the area of the development, production, and utilization of computers.[11] The main outcome of the agreement was the decision to join the Soviets in their effort to manufacture the common computer platform—the Uniform System—in order to, among other objectives, produce a family of high-performing computer systems, compatible across the Soviet bloc.[12] Instead of committing millions of rubles to the research and development of an indigenous high-performing family of compatible

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

Figure 1. Sergey Popov operating his Micro-80 computer at the Moscow Institute of Electronic Engineering in 1979. The computer is connected to a Hungarian-made Videoton-340 terminal. (Photograph courtesy of Sergey Popov.)

mainframes, CMEA settled on cloning the IBM System/360 architecture, obtaining the necessary information through legal channels and covert efforts.[13] The System/360's supremacy in the world's mainframe market and its vast software library made it an irresistible blueprint for the next generation of CMEA mainframes. However, by the time CMEA showcased its first computers of the Uniform System family during the "EC ЭВМ–1973" international exhibit that took place in Moscow in May 1973, almost a decade after the launch of the IBM System/360 family,[14] a new threat appeared on the CMEA high-technology horizon: the large-scale integration (LSI) of semiconductor devices. Novel semiconductor technologies and products, such as the microprocessor, of course made the CoCom's export controls lists.

The far-reaching political, military, and economic ramifications of the rapid progress in semiconductor technologies forced the USSR to promptly find a way to respond to the new technological race. As was the case with the Uniform System program, the Soviets resorted mostly to functional duplication and reverse engineering of Western semiconductor devices, backed by large-scale industrial espionage efforts to acquire advanced IC technology, manufacturing, and test equipment, as well as end products.[15]

The first Soviet LSI CPU chipsets came in the mid-1970s from Zelenograd, the USSR's Silicon Valley. The K587 four-chip bit-slice CPU was used in several Электроника [Electronics]-series computers, but there is no evidence of any significant impact of these early, low-volume Soviet microcomputers on the diffusion of computing in Soviet society.[16] According to some technology analysts, by the early 1980s, there were about 15 Soviet microprocessor chipsets, most of them clones of Western devices, even down to their part numbers.[17] In spite of full recognition of the significance of microprocessor and microcomputer technologies for the future economic and social development of the USSR (given, for instance, during the 1981 XXVI Congress of the Soviet Communist Party) and despite strong state support for the inventor movement, Soviet industry was unable to manufacture state-of-the-art microprocessors and memory chips in quantities that would make them available on the open market and would generate domestic demand for such products outside of military and strategic industrial sectors.[18] That manufacturing incapacity also kept the Soviet electronics hobbyists of the 1970s and early 1980s away from experimenting with the microprocessor.

The First Step

In 1978, the Moscow Institute of Electronic Engineering (MIEE) obtained an early sample of the Soviet KR580IK80 single-chip microprocessor. The device was a functional analogue of the Intel 8080 CPU, replicated at the Kiev Research Institute of Microdevices between 1976 and 1978 and manufactured by the company Кристалл [Cristal].[19] Three MIEE employees (Gennady Zelenko, Victor Panov, and Sergey Popov) decided to experiment with the chip and to build a computer around it. Thanks, in part, to the availability of the Intel 8080 technical literature, they had a working prototype of the computer the following year. They named it Микро-80 [Micro-80] and decided to popularize it (see Figure 1). The state companies and government officials seemed unimpressed with their effort. "We, understanding the potential of microprocessors and microcomputers, contacted various organizations. The big computer firms, the ministries. No one understood us," recollected Popov.[20] Then they turned to *Radio*. "At that time, the *Radio* magazine had a circulation of over 1 million copies, and a readership of a few million people." continued Popov. "I had been reading the magazine since 1966, and so I proposed to my colleagues to contact the editors." The magazine's editor in chief liked the microcomputer theme, possibly because it perfectly fit the prioritization of the microprocessor and microcomputer technologies directive adopted by the 1981 XXVI Congress of the Soviet Communist Party. In September 1982, *Radio* began publishing "The First Step," Zelenko, Panov, and Popov's series of articles demystify the microprocessor and its programming for electronics hobbyists.[21]

The authors provided the schematic diagram and principles of operation of their rudimentary microcomputer for illustrative purposes only. To their surprise, many *Radio* readers decided to embark on the Micro-80 construction project despite the fact that most of the required chips, including the computer's KR580 CPU, could only be purchased on the black market where they occasionally ended up after being stolen from factories and organizations.[22] In hundreds of letters addressed to the authors, hobbyists across the country requested more construction details. All of a sudden, in a country where retail clerks were routinely tallying a sale using pen and paper or an abacus,[23] Soviet electronics enthusiasts wanted to build computers of their own.

The Micro-80 was the first microcomputer project published in the Soviet Union. Although by Popov's own estimate only a few hundred Micro-80s were actually built, the project's publication ignited microcomputer hobby activities, similar to the impact the Mark-8 and the Altair 8800 hobby computer designs had in turning the attention of North American electronics hobbyists to computers in the mid-1970s.[24]

By 1985, the Soviet hobby movement was in full swing. In August of that year, *Radio* published a review of the 32nd All-Union Exhibit of Achievements of Radio Amateurs-Constructors. The review acknowledged that, for the first time, microcomputers stole the show. More than 20 microcomputers were shown by Soviet hobbyists, micros with names such as Альфа-85 [Alpha-85] and АПАС-80 [APAS-80], a KR580-based micro built by members of an amateur radio club Патриот [Patriot].[25]

Still, the movement did not create any network of computer clubs or user groups. As explained by Popov, the Micro-80 computers were assembled mostly by individuals working alone. "It was practically impossible to organize a hobby club and find a place for meetings without permission from the government and party authorities."

The *Radio* magazine itself became the main information hub for computer hobbyists. Through the 1980s, it would publish a variety of articles dealing with microcomputer architectures, programming, and applications. The magazine reported on microcomputer-related events (such as exhibits) and novelties. It initiated debates on the state of Soviet computer literacy programs and hobby movement. In January 1986, the magazine published a synopsis of the roundtable discussion, "Your Personal Computer," which it organized to discuss the obstacles faced by computer hobbyists. The discussion stressed problems such as the persistent unavailability of microprocessors on the open market, the shortage of good quality publications on microprocessors and microcomputers (both on elementary and advanced levels), and the need to acknowledge the contributions of programmers to the computer hobby movement.[26]

The Radio-86RK

The success of the Micro-80's publication paved the way to several other microcomputer construction projects published in mass-distributed magazines. The Ириша [Irisha] educational computer project appeared in 1985 in *Микропроцессорные Средства и Системы* [*Microprocessor Tools and Systems*] magazine, which launched a year earlier under the editorial control of an eminent Soviet computer scientist and proponent of computer education, Andrei Petrovich Ershov.[27] The magazine was the first Soviet periodical exclusively dedicated to microprocessors and their applications.

In 1987, *Моделист–конструктор* [*Modeler-Constructor*] published the Специалист [Specialist] microcomputer construction project authored by an Ukrainian engineer, A.F. Volkov.[28] The ЮТ-88 [UT-88] hobby computer was described in *Юный Техник* [*Young Technician*] magazine in 1989, and in January 1990, *Radio* published the construction details of the Орион-128 [Orion-128].[29] Table 1 lists several early Soviet microcomputer construction projects and hobby computers.

In 1986, Zelenko and Popov published another microcomputer project in *Radio,* coauthored with Dmitri Gorshkov and Yuri Ozerov. Their new Радио-86РК [Radio-86RK] computer was a substantial improvement over the Micro-80 concept.[30] While the Micro-80 unleashed the microcomputer hobby activities, the Radio-86RK lent considerable impetus to the otherwise inept and sluggish Soviet microcomputer industry as several variants of this design were created and turned into volume-manufactured home

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Table 1. Soviet hobby computers, kits, and construction projects most frequently discussed in the Soviet publications of the 1980s.*

Name	Type	Manufacturer/designer	Year of release/ publication
Микро-80 [Micro-80]	Computer project	G. Zelenko, V. Panov, and S. Popov	1982–1983
Вектор-06Ц [Vector 06C]	Personal computer	D. Temirazov and A. Sokolov	1985
Ириша [Irisha]	Computer project	V.N. Baryshnikov et al.	1985-1987
Радио-86РК [Radio-86RK]	Computer project	D. Gorshkov et al.	1986
Электроника КР-01, 02, 03, 04 [Electronics KR-01, 02, 03, 04]	Computer kits	Various manufacturers	1986–1989
Специалист [Specialist]	Computer project	A. Volkov	1987
Кристалл2 [Cristal2]	Personal computer	V. Sugonyako and A. Vinogradov	1987
Ленинград-1 [Leningrad-1]	Single-board computer	S. Zonov	1988
Орион-128 [Orion-128]	Computer project	V. Sugonyako, V. Safronov, and K. Konenkov	1989
ЮТ-88 [UT-88]	Computer project	V. Bartenev	1989–1990

*The list, compiled by the author, is neither complete nor does it include popular clones and refinements of microcomputers included in the table. With the exception of the Leningrad that utilized the Zilog Z80 microprocessor, all listed computers employed the KR580 CPU.

computers by several state companies and the growing home computer cottage industry. The long list of Radio-86RK clones and refinements includes computers such as Альфа-БК [Alpha- BK], Апогей БК-01 [Apogee BK-01], Криста [Krista], Импульс-02 [Impuls-02], Партнер 01.01 [Partner 01.01], and Спектр 001, [Spectrum 001].[31] Furthermore, some electronics stores, such as Moscow's Young Technician, sold 86RK-based computer hobby kits named Радиоконструктор Электроника КР-01, -02, -03, and -04 [Electronics KR-01, -02, -03, -04] put together by various manufacturers.[32]

Perhaps the best-known Radio-86RK refinement was the Микроша [Microsha] designed by the Radio-86RK's authors for mass manufacturing by the Moscow's Лианозовский Электромехнический Завод [Lianozov Electromechanical Company], the same company that in 1984 embarked on manufacturing the АГАТ [Agat] computer, a troubled Soviet clone of the Apple II.[33] In mid-July 1986, a Moscow store and showroom called Радиотехника [Radio-technics] demonstrated the Microsha to citizens. The two-day event brought large crowds of computer enthusiasts, most of whom had never seen a personal computer before.[34]

Soon after the Microsha's launch, its manufacturer opened a computer club in Moscow, making several Microsha and Agat computers available to the general public. Unfortunately, access to these computers was limited to those who could afford the 1.2 ruble/hour fee to work on an Agat or 0.5 ruble/hour to play a game on a Microsha at the time when a monthly salary of a starting engineer was between 120 and 140 rubles a month.[35] Microsha had no direct impact on the movement that brought it to life. It was an expensive personal computer with a hobby blueprint that could be assembled for a fraction of its 550 rubles price tag.[36] However, it was a popular entry-level choice for use at home and school thanks to the support coming from the Radio-86RK community.

The Radio-86RK project culminates the first wave of Soviet computer hobby activities that managed to create a vibrant microcomputer oasis on the barren Soviet personal computer landscape. Fortunately for the movement, the Moscow regime did not consider computer hobbyism to be politically dangerous, an activity that could have facilitated ideologically undesirable activism. On the contrary, its encouragement was exemplified by a large volume of microcomputer-related publications in Soviet youth-oriented magazines. In Popov's opinion, the Communist Party did not object to the popularization of microcomputing outside of the party-approved initiatives:

> These [Party] people saw no threat whatsoever to their dominant position in the information control. What threat can there be from a little

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

> thing [the microprocessor], the size of a finger nail?

Furthermore, the lack of modems and the poor state of Soviet telephone network infrastructure not only isolated the movement internationally but also prevented Soviet microcomputing of the 1980s from creating its own cyberspace of Electronic Bulletin Board Systems and networks, frustrating the adoption of any of the forms of electronic activism practiced in the West.[37]

Enterprising Hackers in the Speccy Land

Although the impact of early hobby designs, such as the Radio-86RK, on the Soviet home computer field was considerable, by the end of the 1980s, the computer hobby movement branched away from the building of indigenous computers to focus on cloning one particular British computer: the Sinclair Research ZX Spectrum (or "Speccy," see Figure 2). The Soviet Speccy era had begun, and by the early 1990s, the Soviet microcomputer scene was dominated by myriad Speccy clones.

Several factors influenced Soviet computer hobbyists' engagement with the cloning of the ZX-Spectrum. The first of them was the dire state of the Soviet electronics industry, which, through the 1980s, was unable to manufacture home and personal computers in quantities that would create the necessary infrastructure for a computer-literate society. In spite of the new information technology-oriented economic plan for 1986–1990 and an ambitious computer literacy program, which, among other objectives, called for a million computers to be installed in Soviet high schools by 1990, the barriers to the mass manufacture of personal and home computers remained intact, allowing only for minuscule production outputs and setting prohibitive retail prices.[38]

The highly publicized Электроника БК-0010 [Electronics BK-0010] computer best illustrates the situation. Released in 1984, the BK-0010 was arguably the first "volume-manufactured" Soviet home computer. It was codeveloped outside of the hobby movement at the prestigious Research Institute of Precision Technologies in Zelenograd and the company Экситон [Exciton].[39] However, due to low manufacturing output, the computer was only available by subscription and only in the Electronics retail stores located in Moscow, Leningrad, and Minsk. This method of sales for household goods was typical of Soviet-era retail. Consumers had to subscribe for a product and wait their turn to get items such as a radio receiver, a freezer, or a TV set. A barebones BK-0010 machine in a box sold for 650 rubles (about half a year's salary for a starting engineer). Without any peripherals or applications software, it was more of a luxurious hobby project than an affordable family computer.[40] According to data published in *Radio,* by mid-1987, only about 2,000 BK-0010's were sold, and although plans for 1987 called for 40,000 computers, only 7,000 were destined for the retail market (the majority of the BK-0010s were to be shipped to schools). None of these production targets were reached. When asked by a *Radio* reporter whether it would be easier to purchase a home computer in the near future, G.P. Morozov, the director of the BK-0010's manufacturer Exciton, replied frankly, "I don't think so."[41,42]

Figure 2. The Sinclair ZX-Spectrum 48K. (Photograph by Bill Bertram, May 2005.)

The large variety of Radio-86RK-inspired products did not solve the home computer inaccessibility problem either because commercial clones of the 86RK could be as expensive as the BK-0010 (for instance, the Partner 01.01 retailed at 750 rubles). Therefore, it is not surprising that in several regions of the USSR computer enthusiasts begun to search for an affordable, under-100-ruble alternative to the BK-0010 and the commercial 86RK clones.

Another factor that turned some computer enthusiasts' attention to the ZX Spectrum in the second half of the 1980s was the lack of software for hobby and commercially retailed home computers. The BK-0010 was sold only with audio cassettes containing samples of Focal and Basic programs, the game *Tetris,* and some test programs. Radio-86RK enthusiasts had to thumb the pages of the 1987 and 1988 issues of *Radio* to get the source code of the Basic interpreter as well as an editor, assembler, and disassembler—programming tools that North American hobbyists were developing more than a decade earlier. The few published games for the Radio-86RK were primitive because the computer did not have graphics or color display capabilities. In short, the BK-0010, the Radio-86RK family, and other early home computers failed to generate a viable personal computer software industry.

At the same time, the microcomputer news coming from the West spoke not of new computer construction projects or better software debugging tools but of vast libraries of game titles, the 8-bit electronic Wonderland accessible through small and inexpensive "software players" such as the ZX Spectrum. This small 8-bit home computer was released by the British company Sinclair Research in 1982. The simplicity of its operation and its low price, good quality graphics and sound, and an extensive software library made it one of the most popular home computers in Europe.

After 1985, the ZX Spectrum could be imported to the USSR because computers of its class were no longer on the export controls lists. However, the computer's price, import duties, and high retail margins made it prohibitively expensive. While paying 1,700 rubles for genuine ZX Spectrum computers was not an option, cloning them was. According to the oral history testimonials of Soviet microcomputer pioneers, some of the ZX Spectrum machines brought to the USSR in the mid-1980s were disassembled and analyzed, and then detailed schematic diagrams were produced.[43] The main challenge in cloning the ZX Spectrum was to find a replacement for its custom uncommitted logic array (ULA) chip that controlled many of the computer's functions such as the display, keyboard, and tape recorder interface. Initially, the "cloners" functionally simulated the ULA chip with the available discrete components. By the early 1990s, Russian clones of that chip as well as of the Zilog Z80 microprocessor employed in the Spectrum were produced.

As in the case of the Radio-86RK, the schematic diagrams of ZX Spectrum clones were in circulation all over the Soviet Union, which resulted in many variants of the Spectrum being put together by hobbyists and state companies. The making and impact of the Leningrad clone designed by Sergey Zonov in 1987–1988 best illustrates that process. The Leningrad was a bare bones version of the ZX Spectrum 48K and, at that time, the simplest among the clones.[44] Like many early computer hobbyists, Zonov was influenced by the Micro-80 computer that he built and experimented with. It was mostly the gossip about the unmatched gaming capabilities of the Spectrum that turned Zonov's attention to the cloning project. He obtained and simplified the ZX Spectrum 48K's schematic diagrams and, after building a few prototypes, distributed his computer's technical information and the printed circuit board layout.[45] Carefully soldering fewer than 50 ICs and a few other electronic components onto the Zonov's Leningrad board produced a rudimentary ZX Spectrum-compatible software player. The Leningrad's design simplicity, ease of assembly, and reliability made it one of the most popular ZX Spectrum clones, resulting in a variety of homebrew and commercial variants of the computer with names such as Composite, Дельта-Н [Delta N], ИТЦ Спектрум [ITC Spectrum], Рита [Rita], Spectrum-Contact, and ZX Spectrum St. Petersburg or with generic names such as Spectrum, Spectrum 48, ZX-Spectrum, and Spectrum Sinclair (see Figure 3).[46]

Initially, the Speccy enthusiasts exchanged information through informal gatherings that frequently took place in the neighborhoods of black markets. "There were no formal clubs," explained Zonov,

> usually we gathered near Young Technician store on [Leningrad's] Krasnoputilovskaya street, on Saturdays and Sundays. It was not a club but it was a place where everybody could find almost any electronic [integrated] circuit, could discuss any technical question. Also it was the main Electronic Black Market in our city. It was a place where we bought CPU, memory, and many other devices. It was a place where it was possible to earn money. For example, I made a device for checking digital chips and memory chips and earned money [that way].

Speccy-related newsletters and magazines, especially those distributed in electronic form on magnetic storage media, came later, in the 1990s. According to data compiled by

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

ZXPRES.RU, their volume peaked in 1997, when almost 400 issues were offered by over 70 of such publications.[47] Compared with the established state published hobby magazines such as *Radio* and *Young Technician,* these electronic Speccy publications were more responsive and welcoming because they were frequently rooted in the Speccy movement itself and had more reader-supplied content.

There was no shortage of genuine ZX Spectrum software on the Soviet market. Those who traveled to East European bloc countries brought back games and other programs cracked by Spectrum enthusiasts from Czechoslovakia and Poland who frequently added their "signatures" in the form of "introductions" (or "demos") to the games.[48] The popularity of these demos and growing interest in the development of domestic games for the clones resulted in the first wave of Russian Demoscene activities, a powerful computer art phenomenon that was born in the Western Europe in the mid-1980s and started to take its roots in Russia half a decade later.[49]

By the early 1990s, the Soviet ZX Spectrum cloning effort was possibly the largest among such undertakings anywhere. The Soviet Speccy machines varied with respect to the degree of compatibility with the British blueprint, and their inclusion of additional features such as built-in Russian fonts or turbo-loading procedures. Many of these computers bore the region of origin's name: Dubna, Krasnogorsk, Novosibirsk, Ural, Leningrad, and Moscow. All these activities clearly infringed on international copyright laws, but as Konstantin Elfimov (who participated in and later documented the Russian Speccy scene) explained, "No one had ever heard the word 'copyright' back then and producing a hacked English computer never counted as a crime."[50]

News of the European success of the ZX Spectrum and of the homebrewed Speccy frenzy eventually reached Zelenograd, where at the beginning of the 1990s, the functional analogues of the Spectrum's ULA chip and the Zilog Z80 CPU were developed.[51] Several state companies capitalized on the Spectrum enthusiasts' passionate efforts to make the home computing widespread and affordable. Using the chips from Zelenograd, they manufactured their own ZX Spectrum clones. These cloning efforts picked up in the early 1990s at a time when the political foundations of the USSR were crumbling and ended only when the Soviet 8-bit clones proved worthless in the new Internet-driven computing reality.

Figure 3. Spectrum-Contact, one of the variants of the Leningrad computer manufactured by the Leningrad-based start-up *Тахион* [Tahion]. The computer was housed in a small, rudimentary plastic enclosure. The computer's keyboard and its glossy stickers were readily available on the Speccy market and used in several other ZX-Spectrum clones. (Photograph by Z. Stachniak.)

Conclusion

The Soviet computer hobby movement took off in the mid-1980s in reaction to the hopelessly muddled process of creating the Soviet version of the Information Age, as a reaction to the stagnant and dysfunctional Soviet economy unable to kick-start a viable home and personal computer industry that was incapable of mass producing even rudimentary home computers and providing them with software. The rise and the impact of the Soviet Speccy suggests that the volume manufacture of computers such as the ZX Spectrum in the mid-1980s would satisfy the needs of computer enthusiasts and most likely confine hobby activities to the niche of electronics hobby clubs. But that did not happen; the low-volume and software-bereft BK-0010s and Microshas had negligible effect on the movement that firmly established itself in the second half of the 1980s.

It is challenging to unequivocally assess the movement's place in the USSR's history of computing. Quantitative data showing the hobbyists' impact on the domestic home computer industry (measured, for

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

instance, by the numbers of hobby-inspired commercial products and their production levels) is neither available nor easy to compile. The same applies to assessing the movement's impact on computer literacy. Clearly, as demonstrated in this article, such impact was broad and far-reaching. The first phase of the movement that focused on self-educating its participants helped to fill pages of popular and technical press with home and personal computer themes, unwittingly stimulating and advancing general interest in personal computing. The hobbyists also offered alternative personal computer designs to those created by the struggling industry and prepared the ground for the emergence of the Soviet Speccy. The second phase was possibly the world's largest undertaking in ZX Spectrum's cloning. It rescued the Soviet home computer industry by offering blueprints for inexpensive to manufacture and easy-to-use software players and by creating strong market demand for them. It also created a vast and dynamic Speccy subculture and the Russian variant of Demoscene born in its wake.

However, the movement's proper assessment cannot exclude areas where the hobbyists were not as successful as they could have been. The movement focused on obsolete 8-bit hardware architectures exclusively and avoided the experimentation with 16-bit microprocessors such as the Soviet K1801BM1 employed in the BK-0010 home computer. It was also unable to create a microcomputer software industry or stimulate the state's interest in launching one. Hence, the movement could not significantly help narrow the Soviet's personal computing gap with the West (measured in terms of the technological sophistication, volume of computers produced, and the levels of penetration of business and educational sectors, among other factors). At best, it could only slow its widening. However, despite these weaknesses, the movement managed to fulfill the unforeseen function of linking the Soviet home computing experience with the Western computing heritage by allowing first-time computer users to experience the 8-bit digital world as fervently and passionately as their Western counterparts had done years earlier when the Apple IIs, Atari 400s and 800s, Commodore VIC-20s and C64s, and Sinclair ZX Spectrums were brought to homes and connected to TV sets for the first time.

Acknowledgments

I owe special thanks to Konstantin Elfimov, Sergey Popov, and Sergey Zonov for patiently answering my many questions regarding the Soviet computer hobby movement and Demoscene. I also wish to express my gratitude to Jaroslav Švelch and Patryk Wasiak for helpful discussion on the East European computer hobby activities of the 1980s and to the anonymous reviewers for their useful comments and suggestions.

References and Notes

1. See A.R. Martin, "Red Hot Apples… or How We Learned to Stop Warring and Love the Pravetz," *inCider*, May 1990, pp. 39–42. Apple II technical specifications, including schematic diagram and firmware, were published by Apple Computer in the *Apple II Reference Manual*, Jan. 1978, chaps. C and D.
2. See, for example, P. Fresiberger and M. Swaine, *Fire in the Valley: The Making of the Personal Computer*, 2nd ed., McGraw-Hill, 2000; S. Levy, *Hackers: Heroes of the Computer Revolution*, O'Reilly Media, 2010; and D. Lungu and Z. Stachniak, "Following TRACE: The Computer Hobby Movement in Canada," *Scientia Canadensis*, vol. 34, no. 1, 2011, pp. 1–2.
3. K. Massie and S.D. Perry, "Hugi Gernsback and Radio Magazines: An Influential Intersection in Broadcast History," *J. Radio Studies*, vol. 9 no. 2, 2002, pp. 264–281.
4. See Fresiberger and Swaine, *Fire in the Valley*; Levy, *Hackers*; and Z. Stachniak, *Inventing the PC: The MCM/70 Story*, McGill–Queen's Univ. Press, 2011.
5. P.E. Ceruzzi, *A History of Modern Computing*, MIT Press, 1998, p. 221.
6. Data from the official *Radio* website, www.radio.ru/archive, as of Nov. 2013.
7. P. Svoren, "ЭВМ: Приглашение к знакомству" [Computer: Invitation to Get Acquainted], *Radio*, no. 3, 1978, pp. 54–56; no. 4, 1978, pp. 51–53; no. 5, 1978, pp. 50–52; no. 6, 1978, pp. 51–53.
8. The subject of computing during the Cold War era has been explored by many authors; see, for instance, P.N. Edward, *The Closed World*, MIT Press, 1997. CMEA was formed in January 1949 and initially included Bulgaria, Czechoslovakia, Hungary, Poland, Romania, and the USSR.
9. The UK's Cold War–era export controls were rooted in its Import, Export, and Customs Powers Act 1939; the United States enacted its Export Control Act in 1949, and Canada its Export and Import Permits Act in 1954. For a

comprehensive examination of export controls, see *Balancing the National Interest: U.S. National Security Export Controls and Global Economic Competition*, Panel on the Impact of National Security Controls on International Technology Transfer (L. Allen Jr. chairman), Nat'l Academy Press, 1987; and D. Burghart, *Red Microchip: Technology Transfer, Export Control and Economic Restructuring in the Soviet Union*, Brookfield, 1992.

10. D.A. Wellman, *A Chip in the Curtain: Computer Technology in the Soviet Union*, Nat'l Defense Univ. Press, 1989; R.A. Stapleton and S.E. Goodman, "The Soviet Union and the Personal Computer 'Revolution,' Report to Nat'l Council for Soviet and East European Research," June 1988.
11. The original signatory countries were Bulgaria, Czechoslovakia, East Germany, Hungry, Poland, and the USSR, with Romania joining the treaty in 1973; see S.E. Goodman, "The Partial Integration of the CMEA Computer Industries, Final Report to National Council for Soviet and East European Research," 16 Aug. 1982, endnote 8, p. 9.
12. The main objectives behind this CMEA undertaking are discussed in Goodman, "The Partial Integration of the CMEA Computer Industry," pp. 10–12.
13. "Soviet Acquisition of Western Technology," report, Central Intelligence Agency, Apr. 1982, pp. 9–10; "Soviet Acquisition of Militarily Significant Western Technology: An Update," report, Office of the Secretary of Defense, Sept. 1985, pp. 4–6.
14. See A.T. Belebcev et al. "Применение вычислительной техники" [Applications of Computational Methods], Энергия, 1977, p. 136.
15. The scope and scale of these efforts are detailed in CIA and DOD reports, "Soviet Acquisition of Western Technology" and "Soviet Acquisition of Militarily Significant Western Technology: An Update.".
16. The early microprocessor activities at Zelenograd are described by B.M. Malashevich, "Разработка вычислительной техники в Зеленограде: первые отечественные микропроцессоры и микроЭВМ" [The Development of Digital Technologies in Zelenograd: The First Domestic Microprocessors and Microcomputers], Электроника: Наука, Технология, Бизнес [Electronics: Science, Technology, Business], no. 7, 2004, pp. 78–83.
17. See, for instance, R. Heuertz, "Soviet Microprocessors and Microcomputers," *Byte*, vol. 9, no. 4, 1984. See also DOD report, "Soviet Acquisition of Militarily Significant Western Technology: An Update," p. 12. In comparison, at the end of 1978, 10 Japanese companies manufactured approximately 80 types of microprocessors, 61 of which were original Japanese designs. See R. Mori et al. "Microprocessors in Japan – Status in 1978," *Computer*, vol. 12, no. 5, 1979, pp. 58–63.
18. For a summary of the socioeconomic framework for the planned Soviet microcomputing age of the 1980s, a summary intended for Soviet electronics hobbyists, see "Микропроцессор – что, где и как?" [The Microprocessor: What, Where, and How?], *Radio*, no. 3, 1983, pp. 34–35.
19. See *Стратегия выбора: 50 лет Киевскому НИИ Микроприборов (1962–2012)* [Strategy of Choice: 50 Years of the Kiev Research Institute of Microdevices (1962–2012)], Y.A. Petin and V.P. Sudorenko, eds., Корнейчук, 2012. The Intel 8080 microprocessor was introduced in 1974 and became the de facto standard CPU of the early North American hobby computers.
20. This and subsequent comments by Popov come from email correspondence between Popov and the author that took place in Feb. 2013. In his "История создания компьютеров 'Микро-80,' 'Радио-86РК' и 'Микроша'" [A History of the Creation of the Computers "Micro-80," "Radio-86RK," and "Microsha"] recollections published at http://zxbyte.ru/history.htm, Popov described a meeting with Nikolai V. Gorshkov, Deputy Minister of Radio Industry (one of the ministries responsible for mainframe and minicomputer production), that took place at MIEE. According to Popov, Gorshkov's response to the Micro-80 demonstration was less than enthusiastic. "Guys, enough of this nonsense. There will be no personal computers. Perhaps a personal car, a personal pension, or a private cottage. Do you really know what a computer is? It is 100 square meters of floor space, 25 service personnel and 30 liters of pure alcohol per month! [for cleaning electrical contacts and connectors]." In 1986, Gorshkov would become the chair of the All-Union Committee on Computer Technology and Information Science established to deal with the unsatisfactory progress of Soviet information and computer technologies. See Wellman, *A Chip in the Curtain*, p. 8.
21. G. Zelenko, V. Panov, and S. Popov, "Первый шаг" [The First Step], *Radio*, nos. 9–12, 1982; nos. 2–4 and 6–11, 1983; nos. 1–3, 1985.
22. In several cities such markets existed near electronics stores, such as Moscow's Pioneer or Leningrad's Young Technician on Krasnoputilovskaya street.
23. Author's recollection of Moscow stores during the 1980 Summer Olympics. See also C.

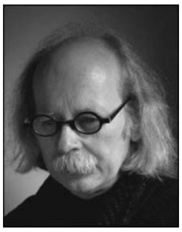
IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Holden, "Soviets Launch Computer Literacy Drive," *Science*, vol. 231, no. 4734, 10 Jan. 1986, p. 109.

24. The Mark-8 and Altair 8800 computers were described in J. Titus, "Build the MARK-8: Your Personal Minicomputer," *Radio-Electronics*, July 1974, pp. 30–33, and E. Roberts and W. Yates, "Altair 8800 Minicomputer, Part I," *Popular Electronics*, Jan. 1975, pp. 33–38.
25. A. Grif, "Диалог с ЭВМ" [Dialogue with a Computer], *Radio*, no. 8, 1985, pp. 2–4.
26. A. Grif and N. Grigorieva, "Твоя персональная ЭВМ" [Your Personal Computer], Radio, no. 1, 1986, pp. 5–7.
27. V.N. Baryshnikov et al., "Персональная ЭВМ «Ириша» для кабинетов информатики н вычислительной техники" [Personal Computer "Irisha" for Computer and Information Technology Classrooms], *Microprocessor Tools and Systems*, no. 3, 1985; nos. 1–4, 1986; no. 6, 1987.
28. A Volkov, "ЭВМ Своими руками!" [Computer with Your Own Hands!], *Modeler-Constructor*, no. 2, 1987, cover, pp. 19–22. The cover of this issue depicts Volcov's prototype of the Specialist, the Фахівець-85, designed by Volkov in 1985 and based on the KR580IK80A CPU. His computer featured graphics (384 × 256) and a color display. The screen of the computer reads, in English translation, "Personal Computer 'Specialist,' designed by A. Volkov … from the city Dniprodzerzhynsk—indispensable helper at work and school, at home and for leisure."
29. UT-88 was described in "Персональный компьютер ЮТ-88" [Personal Computer YT-88], *ЮТ для умелых рук* [UT for Able Hands], supplement, *Young Technician*, no. 2, 1989, pp. 1–16. Orion-128 project was announced in the October 1989 issue of *Radio* and published as V. Sugonyako, V. Safronov, and K. Konenkov, "Персональный радиолюбительский компьютер «Орион-128»" [Personal Hobby Computer "Orion-128"], *Radio*, Jan. 1990, pp. 37–43.
30. D. Gorshkov et al., "Персональный радиолюбительский компьютер «Радио-86РК»" [Personal Hobby Computer Radio-86RK], *Radio*, no. 4, 1986, pp. 24–26; no. 5, 1986, pp. 31–34; no. 6, 1986, pp. 26–28; no. 7, 1986, pp. 26–28; no. 8, 1986, pp. 23–26; no. 9, 1986, pp. 27–28.
31. The relationship between these computers and the Radio-86RK has been established by examining these computers' motherboards and using information provided by S. Popov in *Персональный компьютер МИКРОША и его программое обеспечение* [Personal Computer Microsha and Its Software], Научно-технический кооператив "ЭМИС" [Scientific and Technical Co-Op], Moscow, 1990. The Alpha-BK, Apogee BK-01, Krista, Impuls-02, Partner 01.01, and Spectrum 001 computers were manufactured by, respectively, Ильичевский завод Квант, Импульс (Поселок Лесной, Тульская обл.), Муромский завод радиоизмерительных приборов (МЗ РИП), Импульс (Ильичевск, Одесская обл.), Рязанский завод счетно-аналитических машин («САМ»), and Орловский завод управляющих вычислительных машин им. К.Н.Руднева. Data was collected from these computers' manuals, product labels, and manufacturers' websites.
32. S. Popov to Z. Stachniak, email, Feb. 2013.
33. S. Popov, "A History of the Creation of the Computers Micro-80, Radio-86RK, and Microsha," and V. Aleksev, "Два сапога - не пара, Или Домашняя «Микро»-история" [Two Shoes Do Not Make a Pair, or Home "Micro"-Story], *Техника–Молодёжи* [Technology for Youth], no. 6, 1991, p. 36. For information on Lianozovskiy Electromechanical Company, Agat, and Microsha, see *Наука и жизнь* [Science and Life], no. 10, 1984, pp. 67–70; no. 5, 1987, pp. 54–55; and no. 7, 1987, pp. 30–32. Agat's early shortcomings are discussed in L.D. Bores, "Agat, A Soviet Apple II Computer," *Byte*, Nov. 1984, pp. 134–136, 486–490. According to data reported by *Radio*, in 1989 fewer than 800 Agat machines were manufactured per month; A. Grif, "«КОРВЕТ» терпит кораблекрушение?" [Corvet Shipwrecked?], *Radio*, no. 12, 1989, pp. 2–4.
34. "«Микроша»" [Microsha], *Radio*, no. 9, 1986, p. 26.
35. A. Boeko, "компьютерный клуб" [Computer Club], *Science and Life*, no. 5, 1987, pp. 54–55. The income data quoted after B.M. Malashevich, "Зеленоградские бытовые и школьные компьютеры" [Zelenograd's Home and School Computers], *Electronics: Science, Technology, Business*, no. 7, 2008, p. 98.
36. In his "Two Shoes Do Not Make A Pair, or Home 'Micro'-Story," Aleksev reports that, in 1987, someone could built a Radio-86RK computer for as little as 40 rubles, compared with commercial Electronics KR series of do-it-yourself kits with only 16 Kbytes of memory and priced at 390 rubles. "Tovarisch salesclerk, how much a kilobyte?" asks Aleksev ending his article.
37. According to data provided by Robert W. Campbell on the Soviet telephone infrastructure in the mid-1980s, "only 23 percent of urban households and only 7 percent of rural households had telephones." Of the 15 million persons on the waiting list for telephones, "8 million have been waiting for ten years or more." R.W. Campbell,

Soviet and Post-Soviet Telecommunications: An Industry under Reform, Westview Press, 1995, p. 16. PeaceNet (founded in San Francisco in 1986) and GreenNet (founded in London in 1985) are two examples of influential, international computer communication networks created to support autonomous social and political activism. See J.D.H. Downing, "Computers for Political Change: PeaceNet and Public Data Access," *J. Comm.*, vol. 39, no. 3, 1989, pp. 154–162. See also GreenNet's history page at www.gn.apc.org/about/history.

38. Soviet computer educational programs and microcomputer manufacturing problems of the 1980s were discussed by, among many authors, Holden, "Soviets Launch Computer Literacy Drive"; Wellman, *A Chip in the Curtain*; A. Grif, "Corvet Shipwrecked?"; and A. Grif, "«КОРВЕТ» на мели, кто виноват?" [Corvet on Sandbank, Who Is to Blame?], *Radio*, no. 7, 1988, pp. 2–4. See also R.A. Stapleton and S.E. Goodman, "The Soviet Union and the Personal Computer 'Revolution,'" report, Nat'l Council for Soviet and East European Research, Univ. of Arizona, June 1988.
39. The design history of the Electronics BK-0010 is detailed by Malashevich, "Zelenograd's Home and School Computers," pp. 96–107.
40. The improved model BK-0010.01, announced in 1986, was even more expensive at 724 rubles. Data quoted from Malashevich, "Zelenograd's Home and School Computers," p. 98.
41. See A. Lukshin, "Тернистый путь БК в наш дом" [Thorny Path of the BK to Our Homes], *Radio*, no. 6, 1987, pp. 6–7. According to Malashevich, between 1983 and 1992, Exciton and other companies manufactured about 162,000 BK-series computers (Malashevich, "Zelenograd's Home and School Computers," p. 100). To put this number in proper perspective, the reader should note that in 1982, Sinclair Research sold over 200,000 of its ZX81 home computers in just a few months. Between April 1982 and August 1983, the company sold approximately 500,000 of its Spectrum ZX computers. See "Personal Computer Breakthrough!" ad in *Science Digest*, May 1982, p. 35, and "Spectrum Sales Top 500,000," *Sinclair User*, no. 19, Oct. 1983, p. 15.
42. Another large-scale microcomputer manufacturing fiasco was the Корвет [Corvet] school system. The Soviet computer literacy drive initiated in 1985 called for the installation of thousands of school computers a year, the majority of which were to be the Corvets. The production rates for the Corvet were set at rather modest levels: 10,000 units were to be shipped in 1987, 84,000 in 1989, and 120,000 by 1990. None of these targets was met. During the first nine months of 1989, the entire industry managed to produce fewer than 30,000 computers for schools, among them only 20,500 Corvets, far short of the 84,000 target. See Grif, "Corvet on Sandbank, Who is to Blame?", and Grif, "Corvet Shipwrecked?"
43. See, for instance, "Так кто же первый? - Когда и как появился Первый Отечесвенный Спектрум?" [So, Who's First? When and Where Did the First Domestic Spectrum Appear?], interview with Y.D. Dobush by V. Klimus, ZXPRESS, 10 Jan. 1998; www.zxpress.ru/article.php?id=636. The ZX Spectrum technical information (including schematic diagrams) was also available in publications such as the *ZX Spectrum Service Manual*, written by Thorn Datatech Ltd. for Sinclair Research, Mar. 1984.
44. S. Zonov and A. Larchenko, interview [in Russian], ZXPRESS, 1 Dec. 1995; http://zxpress.ru/article.php?id=289.
45. S. Zonov to Z. Stachniak, email, May 2013.
46. All these computers employ the Leningrad motherboard.
47. "Хронология выхода электронных газет и журналов на ZX Spectrum" [Chronology of Electronic Newspapers and Magazines for ZX Spectrum], ZXPRESS; http://zxpress.ru/chronology.php.
48. P. Wasiak, "Playing and Copying: Social Practices of Home Computer Users in Poland during the 1980s," *Hacking Europe: From Computer Cultures to Demoscenes*, G. Alberts and R. Oldenziel, eds., *Springer*, 2013.
49. See Alberts and Oldenziel, eds., *Hacking Europe*; L. Tasajärvi, ed., *Demoscene: The Art of Real-Time*, Even Lake Studios, 2004; K. Elfimov, "Brief History of Russian Speccy Demoscene and the Story of Inward," *Numerot, Demoscene & Paris Art Scene*, vol. 31, 2008; www.mustekala.info/node/35580.
50. K. Elfimov to Z. Stachniak, email, July 2012.
51. For more detailed account of the development and application of the Soviet analogues of the ULA and Z80, see Malashevich, "Zelenograd's Home and School Computers."



Zbigniew Stachniak is an associate professor in the Department of Electrical Engineering and Computer Science at York University, Canada, and the curator of York University Computer Museum. His research interests include artificial intelligence, history of computing, and history of mathematics. Contact him at zbigniew@cse.yorku.ca.

History of Computing in India: 1955–2010

Vaidyeswaran Rajaraman
Indian Institute of Science

The history of computing in India is inextricably intertwined with two interacting forces: the political climate and government policies, mainly driven by the technocrats and bureaucrats who acted within the boundaries drawn by the political party in power. Four break points occurred in 1970,1978,1991, and 1998 that changed the direction of the development of computers and their applications and affected the growth of IT in India.

The history of computing in India began in 1955, when a HEC-2M designed by A.D. Booth was installed at the Indian Statistical Institute (ISI) in Calcutta.[1] In 1955, a team headed by Rangaswami Narasimhan started designing and fabricating a digital computer at the Tata Institute of Fundamental Research (TIFR) in Bombay.[2] At this time, only a small number of scientists and engineers in India knew about computers. In 2010, the year we end our history, there were more than 2.4 million people employed in computer-related jobs and more than 60 million PCs were in use. Information technology (IT) contributed 6.4 percent of the gross domestic product (GDP) of India, and IT services became the fastest growing segment among export industries. IT grew by 22.7 percent in 2010 with aggregate export revenue of US$50 billion and domestic revenue of US$17 billion.[3] Undoubtedly, this was an exciting journey from 1955 to 2010, although not smooth and steady. In the 1960s and 1970s, there was a lot of trepidation about the use of computers and their impact on employment. Questions were asked about whether computers were relevant for an overpopulated, poor country. From this uncertain beginning, India reached a stage in 1998 when the then prime minister of India declared IT as "India's tomorrow." How did India reach this state? What were the contributing factors? What lessons did India learn?

The development of computing in India is inextricably intertwined with two interacting forces: the political climate and the government policies, mainly driven by the technocrats and bureaucrats guided by the political party in power and some external forces. There were four "break points" in the development of computers and their applications in India as a result of significant events that occurred in 1970, 1978, 1991, and 1998. This article explains why these breaks occurred and how they affected the growth of IT in India.

Several studies on various aspects of the development of computers in India are available in the literature. Dinesh C. Sharma's book, *The Long Revolution—The Birth and Growth of India's IT Industry* gives a detailed account of the history of IT in India from a journalist's perspective.[4] In an earlier book, C.R. Subramanian describes computer technology in India before 1990 and highlights the weaknesses inherent in the government policy of planned development of computers.[5] The book has detailed statistics and excerpts from government archives. Ramesh Subramanian traces the history of IT in India by talking to five IT professionals representing different groups—the government, education, research, and industry.[6] Joseph M. Grieco analyzes how India negotiated with the international computer industry to preserve its national interest without becoming subservient to multinationals.[7] Balaji Parthasarathy discusses how India's domestic policy initiatives enabled the Indian software

 Published by the IEEE Computer Society 1058-6180/15/$31.00

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

industry to grow rapidly.[8] Ross Basset analyzes the impact of computer education initiatives taken by IIT, Kanpur, and other IITs and how this mutually helped India and the United States.[9] Peter Walcott and Seymour Goodman document the growth of computer networks in India from 1980 to 2003.[10] In an earlier article, I give a brief retrospective of IT in India during the period 1965–1999.[11] Articles in the first part of the book, *Homi Bhabha and the Computer Revolution,* edited by Rudrapatna K. Shyamasundar and M. Ananth Pai, present a historical perspective on the development of computer technology in India.[12] The book also includes articles written by pioneers of the Indian computer and communication industry and provides a wealth of first-hand material. James W. Cortada analyzed the international diffusion of IT from 1940 to late 1990s.[13] His book includes a chapter on IT diffusion in India from the perspective of a western professional historian.

To the best of my knowledge, no article has been published on the history of IT in India, identifying break points, why they occurred, and how they changed the pattern of growth. This article aims to cover this gap. It summarizes a monograph I wrote that was published on the Web by the IEEE Computer Society.[14] I wrote this article not as a professional historian but as one who participated in the development of IT in India from the 1960s to date as an academic, an IT consultant to a large number of industries, and a member of numerous committees of the Government of India involved in policymaking. Hopefully this article will kindle the interest of professional historians to undertake a deeper study of India's IT history.

Some Facts about India

India is the second most populous country in the world, with a population of 1.2 billion. It is an old civilization but a young country, with a median age of 26.2 years. India has adopted a mixed economy—a number of public sector companies control oil, natural gas, and heavy industries. Currently there is a thriving private sector. IT is dominated by the private sector. Privatization of major industries began in the early 1990s. Economic growth rate was slow (around 3.8 percent of the GDP) for the first 30 years after independence in 1947, but it was above 7 percent between 2002 and 2010. India's current GDP (based on purchasing power parity) is US$4 trillion, the fifth highest in the world,[15] although the GDP per capita is only US$3,500. The country faces long-term challenges—inadequate physical and social infrastructure, widespread poverty, wide disparity in the quality of education offered, limited rural employment opportunities, and waste in government spending.

Communication facilities in India have improved rapidly since the mid-1980s. Currently, India has more than 40 million landlines and 850 million mobile phone subscribers. India builds and launches its own satellites. Unlike the deep penetration of mobile phones, however, that of the Internet is limited. Although the number of Internet users is estimated to be around 100 million (the third largest in the world), the Internet reaches only about 8.4 percent of the population.

A major problem faced by India is an endemic shortage of electricity. Electric power generation is primarily controlled by the states, and inadequate investment in this important sector has adversely affected all industries, including IT.

Manufacturing industry such as automobiles has matured owing to collaboration with Japanese and western companies. Most electrical consumer goods are manufactured locally and their quality is good. Although local manufacturing of PCs and sophisticated ICs is almost nonexistent, there is a thriving industry designing ICs and other sophisticated electronic hardware for western customers. The world class software services industry is quite mature.

Although Hindi is the official language, English, which was introduced by the British, continues to be the language used by the government and the judiciary. The medium of instruction in science and engineering courses in colleges is also English. It is estimated that 125 million people in India have a working knowledge of English.[16] The large number of English-speaking persons in the 20–50 years age group (estimated to be more than 50 million) has led to a number of IT-based services from western countries being outsourced to India.

The dream of Jawaharlal Nehru, the first Prime Minister of India, was to make India a social democracy with a mixed economy—that is, he envisioned the coexistence of private and state enterprises. His government believed in planned development and constituted the Planning Commission. The primary function of this commission was and continues to be the drafting of five-year plans for

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

the growth of various sectors of the economy and to allocate resources (see www.planningcommission.gov.in/aboutus/history). Jawaharlal Nehru had great faith in science and technology as engines of economic growth. He was convinced that India required rapid industrialization to reduce the abysmal poverty of its people. India's Parliament passed the Scientific Policy Resolution in 1958, the full text of which may be found at www.dst.gov.in/stsysindia/spr1958.htm.

Another problem India faced was a balance of payments deficit. Setting up heavy industries demanded imports, and most of the required petroleum products had to be imported. Exports comprised mostly raw materials such as cotton and minerals. There was dearth of foreign exchange (that is, hard currency such as US dollars earned through export), which dictated many of the government's policies. Any private industry requiring import using scarce foreign exchange was subjected to close scrutiny. A company was required to earn foreign exchange through export in order to import goods. After liberalization in 1991, this situation changed and imports of capital goods including computers became easier, and Indian companies are now allowed to invest abroad. However, the rupee is not a freely convertible currency.

Laying the Foundation (1955–1970)

The first group to build a digital computer in India was led by R. Narasimhan at the TIFR, Bombay. This group started building a computer, the TIFR Automatic Computer (TIFRAC), in 1955 and completed it in 1959. It had 2,048 word core memory (40-bit word, 15 microsecond cycle time), an arithmetic unit with some innovations,[2,12] a paper tape I/O, and a cathode ray tube output unit. An assembler was written for it. It was used for solving physics problems in-house. It was also used by the scientists of the Atomic Energy Establishment and some universities. This project proved that Indians could design a computer and use it effectively. In 1959 there were no commercial digital computers in India.

The first computer with a Fortran compiler to be installed in an educational institution in India was an IBM 1620 at the Indian Institute of Technology Kanpur (IITK) in August 1963. It was used to spread computer education in India. Details of the contribution of IITK to the development of IT in India may be found elsewhere.[9,17] The first high-performance computer installed in India was a CDC 3600-160A at the TIFR in mid-1964[18,19] and was used extensively by several universities and research institutes.

During the 1960s, computers were not considered a priority area that deserved foreign exchange outflow. IBM and the British Tabulating Machines (which was later named International Computers Ltd., ICL) were already selling mechanical accounting machines in India. Because importing computers using foreign exchange was difficult, IBM and ICL applied for licenses to manufacture computers in India. IBM started manufacturing punch card machines and exported them. With the foreign exchange earned, it imported used 1401 computers, refurbished them, and rented them to organizations in India. By 1970, the IBM 1401 was the most popular computer in India, with 80 installations. (In 1971, the IBM 1401 was phased out in the United States.) The annual rental charge was higher than it was in the US. In its defense, IBM asserted that it was selling a computing service rather than just renting computers. And to its credit, IBM's service was excellent, and it recruited and trained good technical and sales persons from India. IBM trained and nurtured a whole generation of maintenance engineers and programmers.[5]

Self-Reliant Growth of the Computer Industry (1970–1977)

During the 1950s, electronics was not considered an important industry by the Government of India and there were no specific policy initiatives. After the border skirmish with China in 1962, in which India lost some territory, it was realized that modern electronics and communication equipment were essential in defense preparedness and for the long-range industrial growth of India. The Government of India constituted a committee in 1963 with Homi Bhabha as its chairman to examine the area of electronics in depth and prepare a plan for its development. One of the main recommendations of the committee was to establish an electronics commission (EC) with wide financial and executive powers to make quick decisions to promote electronics and computer industry[20] and the Department of Electronics (DoE) to execute the policies formulated by the EC.

Political Environment and Government Policies

The Congress party, which spearheaded the independence movement, governed India uninterrupted from 1947 to 1977. The public sector had a preeminent role in the economy. All industries were centrally controlled. India had a perpetual balance of trade deficit. The rupee was not convertible. Thus, obtaining hard currency for importing equipment required elaborate justification.

There was also some trepidation regarding the impact of computers on employment. A committee chaired by Member of Parliament Vinayak M. Dandekar examined the issue in 1972 and prescribed strict controls on introducing computers in industry and government departments.[21] Parliament's Public Accounts Committee in its 1975 report was also cautious.[22] The committee opined that the use of computers could lead to efficiency, higher profits, and faster economic growth, but only in the long run. The committee felt that in India, with its large-scale unemployment, the use of computers and other sophisticated machines for labor-saving applications might not be desirable and might even be detrimental. It recommended that the government take into account the social cost of computerization and evolve a principled and positive approach on computerization, keeping in view the overall national interest.

The Department of Electronics was constrained by these observations and had to move cautiously. In the 1970s, the DoE drafted an elaborate set of rules for importing computers, which led to a delay of one to three years in obtaining clearance to import.[23] In a fast-changing field such as computers, this delay was unacceptable. This restriction particularly hurt companies that wanted to import computers for software export. It also adversely affected the projects of many industries, scientific research laboratories, and universities.

In the 1970s, delays due to import restrictions adversely affected the projects of many industries, scientific research laboratories, and universities.

The other major political event that affected the development of computers was the war with Pakistan in 1971, which ended with the creation of Bangladesh. Richard Nixon, the US president, favored Pakistan, and this resulted in embargos on electronics and computer imports from the United States. The first nuclear test by India in 1974 further aggravated Indo-US relations, leading to an embargo on the import of electronic equipment using advanced technology, which included high-end computers and sophisticated software used in science and engineering.

Bhabha Committee's report of 1968 had recommended that computers, other than the larger ones, must be locally manufactured. It opined that "attaining self-sufficiency in systems engineering and fabrication is of fundamental importance from the point of view of the defense and security of our country." It also suggested that, from a long-range perspective, India should upgrade its capability to design and manufacture smaller computers. Consequently, the DoE fully funded a computer division in the public sector company Electronics Corporation of India Limited (ECIL) to manufacture and market computers.[24] Meanwhile, computer technology was rapidly progressing in the west. With the development of large-scale integration (LSI), the price of computers was falling. The DoE was expected to announce a "minicomputer policy" that would formulate the ground rules for private companies to manufacture computers; however, it delayed the announcement because it was concerned about the requirement of hard currency for importing components and peripherals. The DoE was also protecting ECIL, which it had funded.

Government Initiatives

ECIL designed TDC 12, a 12-bit real-time minicomputer, in 1969. Meanwhile, technology was changing. TDC 12 was upgraded in 1974 to TDC 312 and was followed by TDC 316, a 16-bit computer built in 1975. In 1978, ECIL manufactured a microprocessor-based system MICRO 78. ECIL sold 98 computers between 1971 and 1978, mostly to government laboratories and universities.[25] An important contribution of ECIL was providing ruggedized computers for the Indian Air Force's Air Defense Ground Environment Systems (ADGES). Because of their sensitive nature, these systems had to be designed and fabricated with Indian know-how and in institutions located in India. Import would have been difficult because sensitive information had to be revealed to vendors, and India was often subjected to sudden embargos on electronic systems by the US and its allies. These systems were deployed along India's borders to detect intrusion by unfriendly aircraft. Each system used three TDC 316 computers, which were ruggedized with components adhering to MIL specifications. The systems with radars developed by the Defence Electronics and Radar Development Establishment in Bangalore were designed by a team at TIFR, Bombay, and 25 of these were deployed by the Indian Air Force from 1969–1984. The published literature does not mention this important contribution by ECIL (see www.tifr.res.in/~sanyal/national.html). The system could have been modified for use in air traffic control systems, but ECIL did not pursue this opportunity. (A brief history of ECIL appears elsewhere.[26])

The DoE was established with the primary objective of promoting the development of electronics and the computer industry. However, between 1970 and1977, the general perception was that it was playing more of a regulatory role and protecting ECIL. The political environment[21,22] and the scarcity of hard currency forced the DoE to take certain actions, particularly in regulating the import of computers and allowing manufacture by the private sector.

Regardless of this perception, the DoE did play a promotional role in a number of areas. Important initiatives taken by the DoE during this period include establishing the National Center for Software Development and Computing Techniques in 1972, establishing the National Informatics Center in 1975 to assist in the government's e-governance initiatives, funding the Army Radio Engineering Network, establishing regional computing centers with large imported mainframes in some universities, funding the ADGES project described earlier, and establishing the Computer Maintenance Corporation (CMC) to maintain imported computers,[14] including IBM computers left behind by IBM.

Private Sector Enters the Computer Industry (1978–1990)

In 1973 India had enacted the Foreign Exchange Regulation Act (FERA) to conserve foreign exchange. Under this act, foreign companies, except those considered essential, were to dilute equity to 40 percent and take an Indian partner. IBM was asked to fall in line. It refused and left India. (A detailed discussion of IBM's negotiations with the Government of India is available elsewhere.[27]) IBM's decision to leave India and the announcement of a long-pending minicomputer policy opened up the industry for the entry of private entrepreneurs and the end of ECIL's monopoly.

Even though the political environment was choppy during this period, Rajiv Gandhi, who belonged to the post-independence generation, was open minded regarding private enterprise and ushered in significant changes when he became prime minister.

Political Environment and Government Policies

The period from 1978 to 1990 was one of political instability in India. After two unstable coalition governments, Indira Gandhi returned as prime minister in 1981. In 1984, Indira Gandhi was assassinated and her son Rajiv Gandhi took over as prime minister. He won the next general election and was prime minister from 1984 to 1989. In the general election of 1989, the Congress Party lost and a coalition government returned. This coalition was unstable and fell in 1990.

In spite of the political instability, computing progressed at a much faster rate during 1978–1990 than during 1970–1978. The period started with the appointment of a committee, chaired by Mantosh Sondhi,[28] whose task was to review the progress of electronics and computing. The Sondhi committee suggested permitting private manufacturers to manufacture computers. It opined that the major emphasis in the development of the minicomputer/microprocessor industry would be on setting up systems engineering companies, which were not necessarily engaged in the manufacture of central processing units or peripherals.

Two other events had a far-reaching impact on the government policy. At Rajiv Gandhi's insistence, all the clerical chores of the 1982 Asian Games held in Delhi were computerized using locally made computers and software developed by NIC. It was a resounding success and convinced Rajiv Gandhi of the importance of computers. A liberalized policy on minicomputers was announced in 1984 as soon as Rajiv Gandhi became prime minister. It allowed private sector companies to manufacture 32-bit machines, removed constraints on the number of computers a company could manufacture, allowed assembled boards with microprocessors and interface electronics to be imported along with application software, and reduced the import duty.[29] The procedures for importing mainframes were simplified. The policy of allowing import of fully assembled boards changed manufacturing to assembly and system integration.

Software exports were promoted by recognizing software development and services as an industry. This recognition led to many fiscal and tax concessions. The sending of engineers abroad to develop and maintain software for clients at their sites and the profits earned thereby were recognized as "software exports." In 1986, a more liberalized software policy was announced that gave further incentives for software export.[30] Computers and software tools used to develop software for export could be imported duty free. Software developed in India could be exported using communication systems such as satellite and cable.

Government Initiatives

The period from 1978 to 1990 was also marked by a number of initiatives taken not only by the DoE but also by other organizations that accelerated the use of computers. These initiatives included starting undergraduate courses in computer science at IITs; starting a new degree course, the master of computer applications (MCA), to educate college graduates in systems analysis and design; funding the establishment of computer-aided design and computer-assisted management centers at elite educational institutions; funding centers to develop knowledge-based computer systems; starting the Center for the Development of Advanced Computing to develop high-performance parallel computers; allowing the use of satellite communication for software development from India by multinationals; and promoting the use of computers by banks and other financial institutions.

In addition, the initiation of two projects requires special mention. One was the project to computerize the Indian Railways seat/berth reservation system, which began in 1984 and was completed in 1986. India has one of the largest railway networks in the world. In 1984, Indian Railways handled more than 5 million passengers travelling in more than 600 long-distance trains with around 50,000 reservation requests. Passengers had to stand in long queues to obtain reservations. CMC developed a reservation system and implemented it at the New Delhi booking office. It had 50 counters, and customers could go to any counter to get a reservation for any train.[31] This system was a huge success and was highly appreciated by the general public because it saved them enormous time. The entire software effort was by local programmers with no "foreign consultants." The reservation system using computers was an eye opener to the general public because it demonstrated the advantages of using computers. There was an attitudinal change among both the general public and white collar workers about computerization. This was the beginning of the acceptance of computers and the realization that, in a country with large volumes of data to be processed, the use of computers is inevitable. Some bank unions also accepted computerization that they had opposed earlier.

Within two years of the 1984 announcement of the new policy, the growth of computers went up by 100 percent, and the cost went down by 50 percent.

The second major initiative taken by the DoE was the establishment of software technology parks. STPs provided infrastructure such as buildings, work stations, continuous uninterrupted power supply, and satellite communication links to software companies located in these parks. Software companies were able to develop software on the computers of their overseas customers using the satellite communication links. Because the investment required to set up a company in an STP was low, this initiative allowed many small entrepreneurs to enter the software services export business. The first STP was established in Bangalore in 1990. STPs were set up later in many other cities and incorporated as STPI (www.stpi.in), controlled by the DoE.

Consequences of Government Policy

The minicomputer policy of 1978 opened up the computer industry and saw the emergence of a number of technical entrepreneurs who started computer manufacturing operations. Among the most notable were the DCM, HCL, WIPRO, PSI, and ORG systems. Unix was the operating system of choice because it was easily available and was standardized for the computers to be installed in banks (a large market segment) by a committee headed by Chakravarti Rangarajan, who was the deputy governor of the Reserve Bank of India. Thus during 1978–1984 a large number of systems programmers in India became experts in Unix development and use. They innovated and adapted Unix to work on computers with small memory. They also adopted C widely because Unix was written in C. This expertise came in handy when India entered the software services export market in the late 1980s.

The 1984 liberalization policy had a deeper impact. Within two years of the announcement of the new policy, the growth of computers went up by 100 percent, and the cost went down by 50 percent.[5] The growth was due to the liberal import of populated boards. Manufacturers had to only do systems engineering and develop appropriate software. This made PCs affordable and spread the use of computers. Another consequence of the liberalization of 1986 was the entry of many multinational companies that collaborated with the local companies as minority partners.

The liberalization of the import of computers and software enunciated in the 1986 policy gave an impetus to the software export industry. Export earnings were insignificant in 1978, but they increased to US$128 million in 1990. Establishing STPs by the DoE allowed many entrepreneurs to enter the software export business. Permitting Texas Instruments (TI) to open an offshore software development center in Bangalore and allowing TI to link with its Dallas center via satellite communication set a new trend. GE computer services, the Citibank software group, American Express, Cadence, and many others set up software development centers in India to take advantage of lower costs and the high quality of software developers. This period also saw the establishment of two industry groups: the Manufacturer's Association for IT (MAIT) in 1982, and the National Association of Software and Services Companies (NASSCOM) in 1988. A detailed account of the contributions of NASSCOM to the growth of India's software industry is available elsewhere.[32]

Liberalization of the Economy and Software Export Growth (1991–1997)

This period was an exciting one for the IT industry in India. India started with a near default economic situation in 1991 that forced the government to open the economy and remove many of the controls on industry. The rupee was devalued (from Rs.17.5 to Rs.26 per US dollar) and foreign investments and industries were welcomed. This, along with the fortuitous circumstances of the need to fix the Y2K bug, the Euro conversion requirement, and a technically savvy influential Indian diaspora in the US opened opportunities to Indian software companies. This, coupled with the advent of fast satellite communication, the availability of human resources with strong English-language skills, quality consciousness of software companies, and project management expertise allowed the industry to get remunerative software services contracts from the west, particularly the US. Export earnings increased from US$128 million in 1990 to US$1,759 million in 1997.[33] The average annual growth rate of 45 percent of the software services export during this period was spectacular. The industry also provided employment to 160,000 software engineers. Thus, by 1997 there was a lot of optimism about the future of IT in India.

Political Environment and Government Policies

In the 1991 general election, the Congress Party returned to power. The new government faced a difficult economic situation. India was about to default payment of its external debts. The country was bailed out by the International Monetary Fund and the World Bank, which imposed a set of conditions that forced India to liberalize its economy. That liberalization changed the course of the history of computing in India. In the 1996 general election, the Congress Party was defeated. A coalition was formed, but did not last, and there was political instability with another coalition coming to power. Fortunately, successive governments did not meddle with the policies relating to the IT industry. The officials at the DoE provided continuity in the policy framework.

A number of concessions were given to software companies after liberalization in 1991. The import duty on computers used for software export was abolished. Software companies' export earnings were made tax free for 10 years. Multinational companies were allowed to operate in India with 100 percent equity. According to N.R. Narayana Murthy, one of the founders of INFOSYS, a spectacularly successful IT software services company,[34] three other policy changes significantly altered the business environment: easier convertibility of rupees to hard currency, permission to raise capital through initial public offerings (IPOs), and the abolition of duties on imported software tools.

During this period, the government also permitted private software companies to have dedicated satellite links with their overseas customers. The National Telecom Policy 1994 (www.trai.gov.in/Content/telecom_policy_1994.aspx) allowed private companies to enter the telecommunication business. This had far-reaching consequences later, particularly for the mobile communication industry. The devaluation of the rupee enabled Indian software companies to be competitive in selling their services and MNCs to establish branches in India at low cost.

Consequences of Changes in Government Policies

With the changes in the government policies, there was a sudden spurt in the activities of Indian software companies. Earnings from exports, which were around US$128 million during 1990–1991 went up to around US$1.76 billion in 1997–1998,[33] an average growth of 45 percent each year. Indian software companies invented what is known as the global delivery model (GDM) and the 24-hour workday for the IT industry.[34] They became quality conscious and obtained certification from the International Organization for Standardization (ISO) as well as the Software Engineering Institute (SEI) at Carnegie Mellon University. The Y2K and the Euro conversion requirements created a large international market. In fact, the Y2K problem, which was considered a big threat in the mid-1990s, did not become one partly because of the large number of software professionals in India and the US working diligently to fix the bug during the period from 1993 to 1999.[35]

The government policy of allowing foreign direct investment resulted in many multinational firms setting up software development centers in India. For example, American Express established a center in Mumbai in 1994 to carry out back-office functions such as accounts receivable, payroll processing, and inventory control. IBM, which wanted to reenter India, partnered with the Tata group and started TATA-IBM in 1992 with a 50 percent stake. Many other companies, such as HP, Oracle, and GE Capital, began operations in India. The liberalization of communications encouraged more than 200 software services companies to set up private dedicated satellite links with their clients to develop and maintain software for them. The period 1991–1997 was one of double-digit growth.[33]

Rapid Growth of the IT Industry (1998–2010)

The impact of liberalization of the IT industry and the subsequent recommendations of the IT Task Force in 2000 were felt during this period. The emphasis on hardware, computer production, and computer imports, which held center stage from 1955 to 1990, had started gradually shifting to software services during 1991–1997. The lower hardware costs, faster communications, and emergence of STPs during 1991–1997, with subsequent earning of hard currency by software services companies, caused policymakers to shift their focus to software. The passage of the Special Economic Zone (SEZ) Act in 2005 allowed duty-free import of hardware. Income tax exemption on export earnings for 10 years gave an impetus to a large number of companies to set up units for software and services export. By 2007, 257 software/service companies were set up in SEZ.[4] The change in the rules that allowed a substantial part of export earnings by Indian IT companies to be invested abroad led to the acquisition of IT companies in western countries by Indian IT companies. Quite a few Indian IT companies became multinationals, with 10 of them listed in overseas stock exchanges. These companies had more than 400 delivery centers outside India, with a presence in 52 countries. The export earnings of Indian IT companies, which was US$2 billion in 1998, grew to US$50 billion in 2010.[3]

Political Environment and Government Policies

In the 1998 general election, a coalition called the National Democratic Alliance (NDA) with the Bharatiya Janata Party (BJP) as the majority partner, came to power. The BJP-led NDA was proactive as far as the IT industry was concerned. The export earnings of IT companies were growing around 35 percent each year. NDA lost the election in 2003. The Congress Party returned to power as a senior partner of a coalition called the United Progressive Alliance (UPA). The liberalized policies initiated in 1998 continued, and the economic growth in 2007 was 9.4 percent.[36]

By 2010, Indian IT companies were recognized as world class based on their performance.

The UPA won the election again in 2008 was in power until 2014.

Soon after the NDA came to power in 1998, it set up an IT task force to suggest wide-ranging reforms and incentives to the IT industry to achieve a target of export earnings of US$50 billion by 2008. The task force gave 108 recommendations to the government (see http://it-taskforce.nic.in).

The other major action taken during 1998–2010 was the expansion of education in IT-related areas and engineering. Private corporate bodies were allowed to set up universities in 2002.[37] India's parliament passed the IT Act in 2000 to facilitate e-commerce. It was amended in 2008. India was indeed one of the first few countries to pass IT laws.

Status of the IT Industry

By 2010, Indian IT companies were recognized as world class based on their performance. From low-level testing type projects, the major companies graduated to develop end-to-end applications such as processing credit card payments. Instead of charging based on manpower cost plus expenses plus profit, companies were now taking fixed-price contracts to deliver application software of requisite quality in a specified time. Indian companies were no longer competing for software services contracts based on low cost but rather on quality and timely delivery.[34] The greatest advantage Indian IT companies now had was project planning experience and process maturity, as evidenced by their attaining SEI's Capability Maturity Model (CMM) level 5 certification.[38] In 1999, six of the 12 CMM level 5 certified companies in the world were in India. By 2010, more than 400 of the Fortune 500 companies were clients of Indian software companies.[39]

In addition to software development companies, some new businesses that depended on software and fast worldwide communication grew rapidly during this period. These were IT-enabled services (ITeS) and business process outsourcing (BPO). ITeS included tasks such as checking insurance claims, filing income tax returns, medical transcription, remote support to fix software bugs, and manning call centers. The call centers operate 24/7 for worldwide customers and require language proficiency mostly in English and some European languages.

BPO primarily performs back-office work (such as accounts receivable, payroll processing, account reconciliation, and inventory management) for a number of organizations, the largest segment being banks and insurance companies. In 1994, American Express was the first organization to start BPO work in India; GE Capital International Services followed in 1997.[39] The success of these pioneers induced a large number of Indian companies to start BPO centers for foreign clients.

Another significant development in India during this period was the establishment of research, design, and development centers for several multinational companies. The centers were being set up in India to take advantage of the availability of high-quality computer science graduates at a reasonable cost. The availability of good quality office space in metropolitan areas and improved communication facilities was another incentive. The policy change of allowing companies to have 100 percent ownership without needing an Indian partner was vital (remember that IBM left India in 1978 when asked to dilute equity to 40 percent). IBM, which had returned to India in 1992 as an equal partner of the Tata group, bought off the group's share in 1999. By 2010, IBM had approximately 85,000 employees in India, second only to the number of its employees in the US. By 2010, other large multinationals operating development centers in India were Accenture, Cisco, Dell, GE, Motorola, Microsoft, Oracle, Adobe, SAP, Philips, HP, and Google. TI, which had started developing software tools in India in 1985, discovered that the quality of its employees in India was as good as that in other countries. The company started end-to-end design of ICs and followed this by introducing new products from its India center. TI's example is typical of multinationals that came to India to take advantage of the low cost. They changed their routine work to work that required design expertise when they found that the quality of engineers was good, and finally they started innovation centers. For example, by 2008, TI had obtained 309 patents from its India center. Other microelectronics design and R&D centers were also set up in India during this period by companies such as AMD, Philips, Intel, and ST Microelectronics. The multinationals operating in India obtained more than 1,600 patents between 2006 and 2010.[40]

An important impetus for the growth of IT during this period was the entry of venture capitalists and angel investors in sizable numbers. In 2007, 905 deals worth US$5.3 billion were signed in addition to 748 Internet-specific deals worth US$4.6 billion.[41]

On the hardware side, commodity PCs and laptop manufacture slowly faded. These machines were assembled from boards and other parts imported primarily from China and Taiwan and sold by several multinational companies. Indian "manufacturers" such as HCL, WIPRO, and Zenith were also assembling machines with imported kits, but their volumes were low because they could not compete with multinational companies in quality and mass manufacturing capability.

India was, however, active in designing high-performance parallel computers. In 2003, CDAC designed a parallel machine, the Param Padma, which used 248 processors and a proprietary interconnection network. Its peak speed was 992 gigaflops and it was ranked 171 in the top 500 list of high-performance computers in the world.[42] The Tata group's Computational Research Laboratories designed a high-performance computer named Eka in 2007 with peak speed of 172 teraflops and a sustained speed of 133 Tflops. It was rated the fourth fastest computer in the world when it was announced in 2007 and was the fastest in Asia.[43]

E-governance grew rapidly during this period as well. Citizen services such as property registration, property tax payment, and government certification used to be manual, slow, and prone to corruption. Use of computers expedited these services and reduced corruption. The income tax department is also now fully computerized.

Train ticket reservation, which was first introduced in 1986, became Web based in 2006, and by 2010 passengers could reserve their seats on any train and print their tickets at home using their Internet-connected PCs. In fact, by 2010, airlines, bus, theater, and many other tickets could be booked using the Internet (see https://www.irctc.co.in).

All these achievements would not have been possible without educated workers. There was a rapid expansion of engineering colleges between 1998 and 2010. The groundwork for the expansion of IT education was laid in 1980 with the introduction of MCA courses and the expansion of computer science undergraduate courses. All major IT companies had in-house training programs ranging from 12 weeks to six months. The training schedule was gigantic as every major software company was recruiting around 8,000 graduates each year.

Conclusions

What did India learn from history during the period 1955–2010? Can the growth of the IT industry between 1991 and 2010 be sustained?

As a poor country with a low demand for computers, it was strategically incorrect during 1970–1980 to try to design computers starting at the component level and hope to be completely self-reliant. It would have been wiser to spend the available scarce resources in systems engineering and to build computers using subassemblies during the early stages of development. This is particularly true in the area of computers, where wealthier countries were making huge investments in R&D, with consequent fast changes in technology and rapid obsolescence.[44] This was realized only in the mid-1980s and the consequent policy change led to a rapid growth of computer availability. The fear of unemployment that was in the background while making decisions regarding the use of computers in the 1970s was misconstrued. In a country with a huge population and voluminous data-processing requirements, one cannot manage without computers. This was amply demonstrated when the Indian Railways computerized reservations in 1986. This project changed the mindset of both the general public and the politicians on computers. Another policy error was not to invest sufficiently in the manufacture of ICs in the late 1970s and seriously attempt to get technical collaboration from leading manufacturers in this area.

The investment made in higher education paid off handsomely. Graduates of IITs during the 1960s became technology leaders and entrepreneurs not only in India but also in the US in the 1980s. The DoE along with the Ministry of Human Resources Development took the initiative in the early 1980s to increase the human resources availability by starting bachelor's degree courses in computer science and MCA courses. A training program for teachers of computer science was also started in the 1980s. Private companies provided vocational training. The DoE initiated an accreditation program and administered standardized examinations to certify the vocational courses given by private institutions. These early initiatives provided the human resources that enabled the software industry to take off in the 1990s.

Research projects funded by the Government of India and the United Nations Development Program, which included the National Informatics Center, the National Center for Software Technology, the Center for Development of Advanced Computing, the Computer Aided Design Centers, the Centers for Knowledge Based Computer Systems Development, and the Education and Research in Computer Networking, among others, created a large pool of technology leaders and strengthened institutional infrastructure. Compared with the investment in education and research, the investment made in computer manufacture by the government companies did not have the same multiplier effect. The investment, however, did meet some strategic requirements in defense and atomic energy.

A slew of liberalized policy initiatives taken in the mid-1980s and the early 1990s led to an exponential growth of IT companies in India. The emphasis on quality certification, systematizing application software development processes, and project management were all essential ingredients for the success of the Indian software services companies in the international market.

IT diffusion in India has been primarily in the urban areas among the middle class population. Most public services offered by the state and central government are now computerized. Banking is also computerized. Diffusion of IT to rural areas, where more than 70 percent of Indians live, has been spotty because of the poor availability of electricity, high cost of computers, and lack of availability of local language-based applications. On the other hand, almost everyone in rural areas uses mobile phones because the cost and usage charges are among the lowest in the world,[45] they do not require continuous availability of electricity, and their perceived value is high.

To sustain growth, the Indian software industry should invest more in R&D and develop innovative products not only for international markets but also for the huge local market, which has remained unexplored. The industry should also explore innovative applications based on mobile phones for the local market.

Acknowledgments

I thank David Walden for suggesting that I write the monograph on which this article is based and for guiding me through the various processes for publication in the *IEEE Annals of the History of Computing*. I also thank Paolo Montuschi for encouraging me to write this article and his assistance in submission. I thank all the anonymous reviewers for their numerous suggestions to improve this article.

References and Notes

1. M. Mukherjee, "The First Computer in India," *Computer Education in India: Past, Present and Future*, U. Banerjee, ed., Concept Publications, 1996, pp. 13–16.
2. R. Narasimhan, "On the System and Engineering Design of the General Purpose Electronic Computer at TIFR," *Proc. Indian Academy of Science*, vol. 52, section A, 1960, pp. 45–57.
3. *Annual Report*, Dept. of Information Technology, Government of India, 2010–2011.
4. D. Sharma, *The Long Revolution: The Birth and Growth of India's IT Industry*, Harper-Collins India, 2009.
5. C.R. Subramanian, *India and the Computer: A Study of Planned Development*, Oxford Univ. Press, 1992.
6. R. Subramanian, "India and Information Technology: A Historical and Critical Perspective," *J. Global Information Technology Management*, vol. 9, no. 4, 2006, pp. 28–46.
7. J.M. Grieco, "Between Dependence and Autonomy: India's Experience with International Computer Industry," *Int'l Organization*, vol. 36, no. 3, 1982, pp. 609–632.
8. B. Parthasarathy, "Globalizing Information Technology: The Domestic Policy Context for India's Software Production and Exports," *Iterations: An Interdisciplinary J. of Software History*, vol. 3, May 2004, pp. 1–38.
9. R. Bassett, "Aligning India in the Cold War Era: Indian Technical Elites, the Indian Institute of Technology at Kanpur and Computing in India and the United States," *Technology and Culture*, vol. 50, Oct. 2009, pp. 783–810.
10. P. Walcott and S. Goodman, "Global Diffusion of the Internet 1—India: Is the Elephant Learning to Dance?" *Comm. Assoc. Information Systems*, vol. 11, 2003, pp. 560–646.
11. V. Rajaraman, "Information Technology in India—A Retrospective (1965–1999)," *Computer Soc. of India Comm.*, vol. 23, no. 7, 2000, pp. 9–14.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

12. R.K. Shyamasundar and M.A. Pai, *Homi Bhabha and the Computer Revolution*, Oxford Univ. Press, 2011, pp. 3–16.
13. J.W. Cortada, *The Digital Flood*, Oxford Univ. Press, 2012, pp. 491–545.
14. V. Rajaraman, "History of Computing in India (1955–2010)," IEEE Computer Soc., 2012; www.cbi.umn.edu/hostedpublications/pdf/Rajaraman_HistComputingIndia.pdf.
15. "The World Fact Book: India," Central Intelligence Agency; https://www.cia.gov/library/publications/the-world-factbook/geos/in.html.
16. "India Speak: English Is Our 2nd Language," *Times of India*, 14 Mar. 2010; http://timesofindia.indiatimes.com/india/Indiaspeak-English-is-our-2nd-language/articleshow/5680962.cms.
17. Shyamasundar and Pai, *Homi Bhabha and the Computer Revolution*, pp. 93–103.
18. R. Narasimhan, "Men, Machines and Ideas: An Autobiographical Essay," *Current Science*, vol. 76, no. 3, 1999, pp. 447–454.
19. Shyamasundar and Pai, *Homi Bhabha and the Computer Revolution*, p. xxvi.
20. "Report of the Electronics Committee," Bhabha Committee, Government of India, 1960.
21. "Report of the Committee on Automation," Ministry of Labour and Rehabilitation, Government of India, 1972. See Subramanian, *India and the Computer*, pp.13–14.
22. Public Accounts Committee, "Computerization in Government Departments: Department of Electronics 1975–1976," Fifth Lok Sabha, 221st report, Apr. 1976. See Subramanian, *India and the Computer*, pp. 14–15.
23. "Case History of Computer Import," report of the Rev. Committee on Electronics, Government of India, Sept. 1979, Appendix 12.6.
24. Shyamasundar and Pai, *Homi Bhabha and the Computer Revolution*, pp. 107–118.
25. O. Vikas and L. Ravichandran, "Computerization in India: A Statistical Review," *IPAG J.*, vol. 6, no. 3, 1978, pp. 309–352.
26. Shyamasundar and Pai, *Homi Bhabha and the Computer Revolution*, pp. 50–63.
27. Subramanian, *India and the Computer*, pp. 161–176.
28. Virendra Kumar, "Committees and Commissions in India: 1978," vol.16, part 1, pp. 662–669.
29. Subramanian, *India and the Computer*, pp. 339–345.
30. Subramanian, *India and the Computer*, pp. 352–362.
31. Sharma, *The Long Revolution*, pp. 144–156.
32. K. Karnik, *The Coalition of Competitors*, Collins Business, 2012.
33. S. Bhatnagar, "India's Software Industry," *Technology Adaptation and Exports: How Some Developing Countries Got It Right*, V. Chandru, ed., World Bank, 2006, pp. 95–124.
34. See Shyamasundar and Pai, *Homi Bhabha and the Computer Revolution*, pp. 151–194.
35. L. Suri, "How India Solves the Y2K Problem," http://pib.nic.in/feature/fe0899/f1508995.html.
36. *Annual Report 2007–2008*, Department of Commerce, Government of India, New Delhi.
37. "Private Universities, University Grants Commission, India," http://www.ugc.ac.in/privatuniversity.aspx.
38. M.C. Paulk et al., "Capability Maturity Model for Software, Version 1.1," Software Eng. Inst., Carnegie Mellon Univ., 1993.
39. G. Rastogi and B. Pradhan, *Offshore: How India Got Back on the Global Business Map*, Penguin India, 2011.
40. R. Basant and S. Mani, "Foreign R&D Centres in India: An Analysis of Their Size, Structure and Implications," working paper 2012-0106, Indian Inst. of Management, Ahmedabad, 2012.
41. D. Verma, "Venture Capital," slide presentation, 2009; www.slideshare.net/deepakverma/venture-capital-2012421.
42. Centre for Development of Advanced Computing, "Param Padma," www.cdac.in/index.aspx?id=parampma.
43. "Tata's Supercomputer Eka Named the Fastest in Asia, 4th in World," *The Economic Times*, 13 Nov 2007; http://articles.economictimes.indiatimes.com/2007-11-13/news/27672171_1_supercomputer-computational-research-laboratories-eka.
44. Subramanian, *India and the Computer*, p. 7.
45. C. Li and B. Ninan-Moses, "An International Comparison of Cell Phone Plans and Prices," New America Foundation, 14 Oct. 2010.

Vaidyeswaran Rajaraman is an honorary professor in the Supercomputer Education Research Center (SERC) at the Indian Institute of Science (IISc), Bangalore. He was awarded the Padma Bhushan (the third highest civilian award) by the President of India in 1998 and is a fellow of the Indian National Science Academy, Indian Academy of Sciences, National Academy of Sciences, Indian National Academy of Engineering, Computer Society of India, and the Institution of Electronics and Telecommunication Engineers. Rajaraman has a PhD in electrical engineering from the University of Wisconsin. Contact him at rajaram@serc.iisc.in.

Useful Instruction for Practical People: Early Printed Discussions of the Slide Rule in the US

Peggy Aldrich Kidwell
Smithsonian Institution

Accounts of the appearance and use of various forms of slide rule appear in a variety of 18th and early 19th American printed sources. Based largely on British work, these texts reveal both confidence in the practical potential of the instrument and the slow diffusion of the device.

The Scottish mathematician John Napier of Murchison published his account of logarithms 400 years ago.[1] Within a few decades, instrument makers had designed and made slide rules, computing devices that used the function. However, the instrument did not spread quickly or widely. An examination of printed sources suggests how diffusion did take place in the first half of the 19th century in the United States. It was only at the end of that century, with the spread of professional engineering and the invention of new forms of slide rule, that the device became common among technical people.[2]

Knowledge and use of the slide rule spread slowly in the early US. In the antebellum years, perhaps two dozen books published in the US included a brief discussion of some form of slide rule.[3] These were practical treatises for working men, designed to fit specific needs of lumbermen, tax collectors, engineers, and businessmen. They were not texts for classroom study. Some of these were pirated editions of English books; others were by American authors who usually drew heavily on English sources. The publications presented procedures that readers could follow to get the answers they sought, not general mathematical principles. Thus, early American users of the slide rule probably had no more sense of why it worked than most present-day users of electronic calculators understand the operating principles of those devices.

The Slide Rule as Part of Practical Mathematics

The first discussions of slide rules were in general texts. In 1799, the New Hampshire mathematics teacher Ezekiel Little (1762–1840) published a short volume with the lengthy title: *The Usher. Comprising Arithmetic in Whole Numbers; Federal Money; Decimal and Vulgar Fractions; A Description and Use of Coggeshall's Sliding Rule; Some Uses Of Gunter's Scale; Superficial and Solid Measuring; Geometrical Definitions and Problems; Surveying; The Surveyor's Pocket Companion, or Trigonometry Made Easy; A Table Of Sines; A Table Of Tangents; Miscellany, Tables of the Weight and Value Of Gold Coins. Calculated and Designed for Youth*. In an introductory message for his readers, Little explained that most previous authors on arithmetic, measurement, trigonometry, or surveying confined themselves to only one of these topics, thus requiring the purchase of several books to complete a child's education. To economically provide youth with "so much knowledge in arithmetic and measuring as is necessary in the common business of life," Little had selected from other authors the practical rules he deemed sufficient for ordinary affairs. His title indicates the range of topics he thought useful. Little urged readers to send him corrections. For the motto on the title page of the book, he selected a couplet from Alexander Pope's *An Essay on Criticism*: "Whoever thinks a faultless piece to see, Thinks what ne'er was, nor is, nor e'er shall be."[4] Despite the author's openness to improvement, a second edition of the book never appeared.

Little's description of the sliding rule was brief but thorough. His principal concern was the carpenter's rule, a sliding rule introduced by the Englishman Henry Coggeshall

 Published by the IEEE Computer Society 1058-6180/15/$31.00

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

(1623–1690) for use in calculations relating to timber (see Figure 1). Little offered examples of how to use the instrument for multiplying, dividing, finding square roots, and squaring numbers. He noted when one might need to multiply or divide by a factor of 10 when the answer was larger or smaller than the length of the line on the rule allowed. He made no mention of logarithms.[5]

Another, more influential compendium to include a brief discussion of the slide rule was Nathaniel Bowditch's *New American Practical Navigator.* Bowditch (1773–1838), a navigator, self-taught astronomer, mathematician, and actuary,[6] first prepared an American edition of John H. Moore's *Practical Navigator* in 1799. The Englishman Moore did not mention the slide rule but had a brief discussion of another instrument with logarithmic scales, Gunter's scale.[7] Bowditch followed this example in 1799 and in the 1802 version of the book, published under his name as *The New American Practical Navigator.* He did, like Moore, include a detailed account of logarithms and their use in calculation.[8]

Bowditch's *Navigator* proved a classic, and it would be published in revised form into this century. The next edition appeared in 1807. Here Bowditch did include a brief discussion of the "sliding rule," noting its possible use in solving basic problems of multiplication, division, trigonometry, and measurement. The section came before the discussion of logarithms. Bowditch's discussion of a slide rule with trigonometric scales was unusual, although not unprecedented. Rules with such scales were known in Britain and discussed in contemporary publications.[9] A fellow resident of Salem, watchmaker James Dalrymple, advertised that he sold a sliding rule that followed Bowditch's design.[10]

Bowditch firmly believed that logarithmic tables offered more useful solutions for a navigator's calculations than a slide rule. To use his words, the "sliding rule is rather an object of curiosity than of real use, as it is much more accurate to make use of logarithms."[11] A section on the sliding rule—although not this comment—would appear in editions of the *Navigator* through at least 1861, well after Bowditch's death.[12] By 1888, discussion of the sliding rule disappeared altogether, although Gunter's scale remained.[13]

While Bowditch's discussion of the sliding rule may have been the most widely published in the antebellum United States, sliding rules did not appeal widely to navigators. They were more generally considered in treatises on applied geometry and measurement or, to use the 19th century term, *mensuration.* One of the most thorough of these discussions was also the first, an 1801 Philadelphia reprint of a book on mensuration by William Hawney, as revised by another English textbook author, Thomas Keith.[14] Hawney (active 1710–1750) had brought together principles of measurement in a volume he called *The Compleat Measurer,* first published in 1717. An advertisement at the beginning of the 1721 edition of this work indicates that the author taught mathematics and the use of instruments at Lydd in the county of Kent. A man of many talents, Hawney also surveyed land and sold garden sundials, pocket dials, and quadrants. Rules are not mentioned among his products. *The Compleat Measurer* included a detailed discussion of arithmetic, both as carried out by hand and as done with the use of Gunter's scale and a set of dividers. In his preface, Hawney referred readers interested in sliding rules to other authors. In the section on gauging (finding the volume of barrels), he alluded briefly to a sliding rule, but did not describe it specifically.[15]

By the closing years of the 18th century, *The Compleat Measurer* had gone through some 16 editions. London mathematics teacher and textbook author Thomas Keith undertook a new edition of the book, condensing some of Hawney's examples and

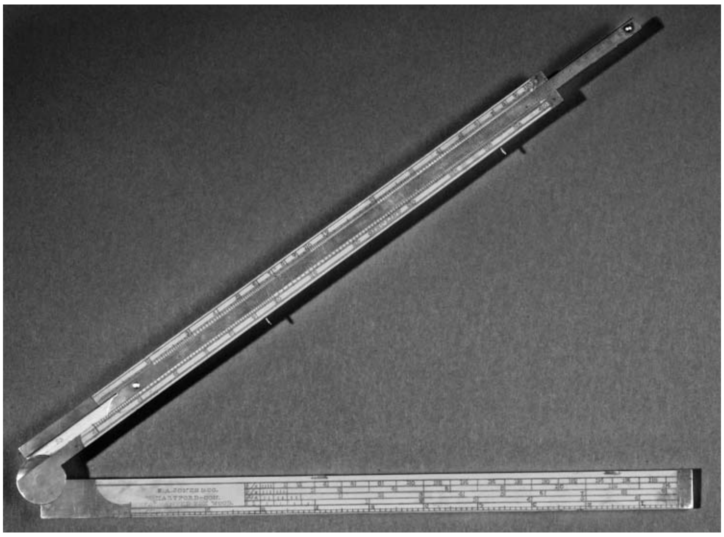


Figure 1. Carpenter's Rule by S.A. Jones and Company of Hartford, Connecticut, circa 1840. The slide extends from the top of the rule. Rules of this type included a girt line, used to estimate the number of board feet in standing trees and logs. (Smithsonian Institution, image number DOR 2014-05429.)

Figure 2. Four-sided Gauger's Slide Rule by Laban Cook of London, circa 1830. (Smithsonian Institution, image number 80-17938.)

Figure 3. Routledge's Engineer's Rule by Thomas Aston of Birmingham, England, circa 1830. Such rules lack the girt line found on the carpenter's rule but have an additional logarithmic scale suited to calculations of squares and square roots, as well as appropriate tables to assist in calculations. (Smithsonian Institution, image number 82-6212.)

adding new material.[16] New sections—carefully marked with an asterisk in the index—included ones on Gunter's scale and on the carpenter's rule. The comments on the gauger's rule (see Figure 2) also were greatly expanded. Keith even noted that such rules could be purchased from the mathematical instrument maker Jones. Moreover, in the course of the text, Keith described how specific problems could be solved not only with Gunter's scale and compasses but with these slide rules. Like Hawney before him, Keith made no attempt to explain logarithms.[17]

Keith's version of Hawney went through at least four editions in the United States. The 1801 Philadelphia edition was followed by a second edition in 1807.[18] In 1813, John D. Craig, a British-born mathematics teacher and textbook author in Baltimore, edited a third American edition, which was republished in 1820. Craig saw no reason to alter Keith's comments on slide rules.[19]

Following in this tradition, in 1805 James Thompson published in Troy, New York, a small volume entitled *A Complete Treatise on the Mensuration of Timber*. Thompson's book included a long discussion of decimal arithmetic and a brief account of the use of the carpenter's rule. He gave no general description of the rule but claimed to offer a "new, expeditious and very accurate method of calculating the contents of square and round timber."[20]

One source Thompson used was the English mathematician Charles Hutton (1737–1823). Hutton was a schoolmaster in Newcastle and then, from 1773 to 1897, a professor of mathematics at the Royal Academy in Woolwich. Hutton published two books on the mathematics of measurement: *A Treatise on Mensuration* (Newcastle, 1770) and *The Compendious Measurer* (London, 1786). The second and shorter of these was reprinted in Philadelphia in 1807. Like Keith's version of Hawney, *The Compendious Measurer* included a brief but relatively thorough discussion of both the carpenter's and gauger's rules.[21] Not long thereafter, yet another Philadelphia publisher offered a briefer account of the carpenter's slide rule in a reprint of another English book, John Bonnycastle's *An Introduction to Mensuration and Practical Geometry*. Like Hutton, Bonnycastle (1750?–1821) taught mathematics at Woolwich for a few years. His book is noteworthy in that, unlike Hutton's *Compendius Measurer,* it would be printed in the US more than once. At least 13 printings of the book appeared in Philadelphia between 1812 and 1849.[22]

The editions of Bonnycastle published in the US described the two-fold carpenter's rule and, in an appendix, the four-sided gauger's rule. From at least 1823, English editions of his book described what they called a "carpenter's rule" that had rather different scales, namely those of Routledge's engineer's rule (see Figure 3).[23]

Bonnycastle's use of engineering scales reflects contemporary British practice and a new role for the slide rule in steam engineering. By 1840 steam engines were also in use throughout the US, powering steamboats and locomotives as well as mills and refineries.[24] Not surprisingly, US rule manufacturers began to sell Routledge's engineer's rule as well as the carpenter's rule.[25] Published accounts followed suit. As early as 1828, rule-maker George Piper, then of Charlestown, Massachusetts, arranged for the publication of the first American edition of Routledge's account of his instrument.[26] This would be republished by another American maker of the instrument, Belcher Brothers, in

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

1844.[27] In 1846 Thomas H. McLeod (1823–1910) published in Middlebury, Vermont, a small book entitled *Instrumental Calculation or a Treatise on the Sliding Rule.* McLeod was born in Elizabethtown, New York. He attended Middlebury College and remained in Middlebury as a lawyer and justice of the peace. At the time, the town had a large marble works and served as a market for timber and agricultural products.[28] McLeod envisioned the volume as suited to the residents of such an industrious community; according to the title, it was "adapted to the ready comprehension of the mechanic, merchant, and farmer, for whom it is designed." McLeod followed earlier authors in presenting procedures for the use of slide rules, with no attempt to explain the underlying principles. His book, like a few others published in the 1840s, is noteworthy in that it discussed only the slide rule and not measurement generally. McLeod also distinguished clearly between the "common slide rule" (the carpenter's rule) and the "engineer's rule" (Routledge's rule).[29]

Also in this tradition is the discussion of the slide rule and instrumental arithmetic in *The Mechanic's, Machinists, and Engineer's Practical Book of Reference…,* first published in New York in 1855 and reprinted at least as late as 1870. This collection of useful information was compiled by railway engineer Charles Haslett and edited by Columbia University mathematics professor Charles W. Hackley (1809–1861). Hackley was an 1829 graduate of West Point who had prepared notes for his students at the University of New York in the 1830s. After spending some time in theological studies, as president of Jefferson College in Mississippi and then as a clergyman in upstate New York, he returned to New York City in 1843 to teach at Columbia. Through these years, he wrote textbooks on trigonometry (1838), algebra (1846), and geometry (1847).[30] These were sufficiently successful for him to be asked to edit Haslett's compendium. At least according to Hackley, Haslett included material not found in other compilations for practical people. However, the discussion of the slide rule followed, word for word, a few pages from William Templeton's *The Operative Mechanic's Workshop Companion and the Scientific Gentleman's Practical Assistant* (London, 1845).[31] It described a slide rule with three identical scales (A, B, and C) and a fourth scale (D) for finding squares and square roots, all logarithmically divided.[32]

The Role of Vendors

Some mid-century authors, like Bowditch earlier, were reluctant to pay much attention to the slide rule. For example, in 1838 Amos Eaton (1776–1842), a senior professor at Rensselaer Polytechnic Institute, one of the first civilian engineering colleges in the United States,[33] wrote *Prodromus of a Practical Treatise on the Mathematical Arts: Containing Directions for Surveying and Engineering.* This small treatise was designed "to enable a common-sense farmer, mechanic, merchant, or other man of business, who is but an ordinary arithmetician, to become sufficiently qualified for the business concerns of life." However it contained only two pages on the slide rule. Eaton, who was an experienced surveyor, explained in the preface that people must be shown the use of instruments "as it is an idle waste of time to learn their use from books."[34] Later in the 19th century, when the slide rule had come to play a considerable role in the engineering classroom, other textbook authors would exhibit a similar reluctance to expound on the slide rule.

Given the hesitation of professors like Eaton, it is not surprising that early American instrument vendors and inventors took pains to write and speak on the possibilities of their instruments. As already noted, Salem, Massachusetts, watchmaker Dalrymple published a broadside some time before 1807 boasting that he sold sliding rules like those described by Bowditch. The republication of Routledge's writings on his engineer's rule by makers of that instrument fits the same pattern. More original writings soon followed. Aaron Palmer of New York state copyrighted a circular slide rule in the early 1840s and published *A Key to the Endless, Self-Computing Scale, Showing Its Application to the Different Rules of Arithmetic, &c.*[35] John E. Fuller (1799–1878) of New York City soon acquired rights to Palmer's instrument, added a second scale for computing the number of days between two dates, and prepared much more extensive instructions on possible uses of the instrument. Mindful of the speed associated with the then-novel telegraph, the inventor dubbed his device "Fuller's computing telegraph" (see Figure 4). It cost $6.00, more than most students, workmen, or teachers could afford in the 1840s. Fuller indicated in the title to the brochure that accompanied the 1852 version of the instrument that it was designed to solve "business questions of every possible variety."[36] His key described these problems in detail. Much in the style of

Figure 4. Fuller's Time Telegraph, London edition, 1860s. With Palmer's computing scale (a circular slide rule on the reverse side), this constituted Fuller's computing telegraph. (Smithsonian Institution, image number 87-4843.)

handbooks of the day, it gave rules for solving numerous types of problems, with examples. Topics ranged from simple multiplication and division to calculations relating to interest and the apportionment of taxes to simple problems of mechanics.[37] Fuller claimed to sell thousands of copies of his computing scale, and collected testimonials from mathematics teachers, businessmen, and bureaucrats in both the US and Britain. He remained in business for more than a quarter of a century. However, his instrument faced competition from less expensive and more portable linear slide rules and published mathematical tables. Production apparently had ended before his death.[38]

Other authors continued to write on more traditional rules. For example, in 1849 Daniel M. Knapen published in New York *The Mechanic's Assistant: A Thorough Practical Treatise on Mensuration and the Sliding Rule*. Knapen's text was devoted entirely to the use of the carpenter's rule and Routledge's rule. It included a drawing of a carpenter's rule made by Belcher Brothers of New York. Knapen also took pains to mention Belcher Brothers engineer's slide rule. He believed that his treatise contained more original matter than almost any other volume on the subject. Indeed Knapen went beyond discussion of square and cube roots to include finding fourth and fifth roots—a task not commonly encountered. Similarly, he talked not only about the measurement of a variety of polygons and circular segments, but he discussed both the computation of areas bounded by conic sections and the volumes bounded by various second-order surfaces (these calculations did not always make use of the slide rule). Knapen expressed the hope that his book would not only serve a practical purpose but inspire readers "to acquire a thorough knowledge of the principles on which the more abstruse rules in this volume are founded," allowing them not only to improve known rules but to deduce new ones. The reaction of mechanics to this assistance is unknown.[39]

Knapen's volume was not published by Belcher Brothers. However, rule makers sometimes published accounts of the slide rule themselves. For example, in 1858 Hermon Chapin, manufacturer of rules, planes, gauges, levels, and the like, published in Pine Meadow, Connecticut, a volume entitled *Instructions for the Engineer's Improved Sliding Rule*…. No author is listed, although Philip E. Stanley reports that this volume is an exact reproduction of the sixth edition of Routledge's *Instructions for the Engineer's Improved Sliding Rule* (Bolton, 1823).[40]

Sometimes publications of American authors were distributed by rule manufacturers. For example, in 1859 Arnold Jillson (born about 1814), a machinist in Woonsocket, Rhode Island, wrote a small volume on instrumental arithmetic devoted to the subject of the carpenter's and engineer's rules. His comments on the second instrument were especially aimed at the cotton mill business, as he hoped to "stimulate the operative to a more thorough knowledge of the changes to be made in his department." Jillson favored brevity over "philosophical minuteness." He was interested in practical matters, not hoping to inspire others to deduce new or more concise rules. His book was published in 1866 by Case, Lockwood, and Company of Hartford and from at least 1872 distributed by Stanley Rule and Level Company of New Britain.[41] The preparation and distribution of instruction manuals on the slide rule and other instruments would become a regular part of the business of slide rule makers and distributors in the late 19th century.

A brief notice of Jillson's book that appeared in *Manufacturer and Builder* in 1874 echoes earlier doubts about relying on

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

instrumental rather than mental arithmetic. It concludes:

> We cannot say that in general we are in favor of calculations obtained by means of such contrivances as sliding-rules, compasses of proportion, or other mechanical instruments; but when we observe to what an astonishing variety of purposes the slide-rule explained in this little volume may be applied, we become reconciled, as it will surely train the mind of many operatives to such a degree that in many instances they will at last be able to make their calculations without help of the rule.[42]

Indeed, with the spread of elementary education in the first half of the 19th century, particularly in the northern parts of the United States, increasing numbers of working people were able to perform simple arithmetical calculations on their own. At the same time, a growing demand for engineering calculations created a whole new market for slide rules.

Conclusion

The slide rule did not become an important part of American mathematical practice until the 1890s.[43] However, thoroughgoing authors—and the occasional maker—discussed the topic in earlier years. Their work suggests the importance of English sources, the aspirations of inventors, and the influence of those who sought both a gain a livelihood from their knowledge of mathematical instruments and to improve industrial practice by better training workers.

Acknowledgments

This article grew out of a research project undertaken with historians of mathematics Amy Ackerberg-Hastings and David Lindsay Roberts. I thank them, National Museum of American History colleagues, numerous librarians, Mark E. Kidwell, and three anonymous reviewers for assistance and helpful comments.

References and Notes

1. J. Napier, *Mirifici Logarithmorum Canonis Descriptio, Ejusque Usus, in Utraque Trigonometria; Ut Etiam in Omni Logistica Mathematica, Amplissimi, Facillimi, & Expeditissimi Explication* [The Description of the Wonderful Canon of Logarithms, and the Use of Which not Only in Trigonometry, but also in all Mathematical Calculations, Most Fully and Easily Explained in the Most Expeditious Manner], A. Hart, 1614.
2. Histories of the slide rule include the classic F. Cajori, *History of the Logarithmic Slide Rule*, Eng. News Publishing, 1919; and D. von Jezierski, *Slide Rules: A Journey through Three Centuries*, R. Shepherd, trans., Astragal Press, 2000. The 400th anniversary of Napier's publication has inspired further research. On the later history of the slide rule in American education, see P.A. Kidwell, A. Ackerberg-Hastings, and D.L. Roberts, *Tools of American Mathematics Teaching, 1800–2000*, Johns Hopkins Univ. Press, 2008, especially pp. 105–122.
3. This discussion of books published in the US mentioning the slide rule is not meant to be exhaustive. I confine myself to books that actually described slide rules in some detail. Thus I do not include textbooks like N. Pike's *A New and Complete System of Arithmetic: Composed for the Use of the Citizens of the United States*, J. Mycall, 1788. Discussing various problems relating to calculating areas and volumes, Pike alludes to a sliding rule. He does not describe it in detail (see pp. 424–426, 428, 430–431, 434, 446–447, 450, 459–460). I mention this work because it is cited by Cajori, *History of the Logarithmic Slide Rule*, p. 59. Other authors, such as B. Stevens, took a similar approach. See, for example, B. Stevens, *A New and Concise System of Arithmetic…*, G. M. Davison, 1822. L.C. Karpinski lists some 321 "tables, ready reckoners, and works dealing with logarithms, the slide rule and mathematical instruments" published through 1850. Most of these say nothing about the slide rule, but his list would be a useful starting point for anyone wishing to create a more precise tally. On the other hand, he does not include the 1850s. See L.C. Karpinski, *A Bibliography of Mathematical Works Printed in America through* 1850, Arno Press, 1980 (this is a reprint of the 1940 edition), pp. 668–672.
4. E. Little, *The Usher. Comprising Arithmetic in Whole Numbers; Federal Money; Decimal and Vulgar Fractions; A Description and Use of Coggeshall's Sliding Rule; Some Uses of Gunter's Scale; Superficial and Solid Measuring; Geometrical Definitions and Problems; Surveying; The Surveyor's Pocket Companion, or Trigonometry Made Easy; A Table of Sines; A Table of Tangents; Miscellany, Tables of the Weight and Value of Gold Coins. Calculated and Designed for Youth*, H. Ranlet, 1799. The quotations are from the first page. The term "usher" refers to a schoolmaster's assistant.
5. Little, *The Usher*, pp. 153–154.
6. For one account of Bowditch's life, see N.I. Bowditch, *Memoir of Nathaniel Bowditch by His Son…*, John Wilson and Son, 1884.

7. J.H. Moore, *The New Practical Navigator…*, B. Law & Son, 1794, pp. 14–16; J.H. Moore, *The New Practical Navigator…*, E.M. Blunt, 1799, pp. 26–28.
8. N. Bowditch, *The New American Practical Navigator…*, E.M. Blunt, 1802, pp. 47–49, 54–61.
9. See Cajori, *History of the Logarithmic Slide Rule*, pp. 32–33. For a British text, see A. MacKay, *Description and Use of the Sliding Gunter in Navigation*, for the author by J. Burnett, 1802. For an historical discussion, see O. van Poelje, "Gunter Rules in Navigation," *J. Oughtred Soc.*, vol. 13, no. 1, 2004, pp. 11–22. One American author who shared Bowditch's interest in rules with trigonometric scales was G. Curtis of Vermont. See his *A Treatise on Gunter's Scale, and the Sliding Rule: Together with a Description and Use of the Sector, Protractor, Plain Scale, and Line of Chords…*, E. Adams, 1824, especially pp. 22–23.
10. "On the Sliding Rule," Broadside, Am. Antiquarian Soc.
11. N. Bowditch, *The New American Practical Navigator…*, E.M. Blunt, 1807, pp. 46–47. The quotation is from p. 47.
12. N. Bowditch, *The New American Practical Navigator…*, E. and G.W. Blunt, 1861, pp. 23–24.
13. N. Bowditch, *The American Practical Navigator…*, rev. ed., P.H. Cooper, ed., Govt. Printing Office, 1888, pp. 3–4.
14. W. Hawney, *Hawney's Complete Measurer, or, The Whole Art of Measuring…*, rev. ed., T. Keith, ed., printed by Cochran and M'Laughlin for M. Carey and J. Thackera [sic], 1801.
15. W. Hawney, *The Compleat Measurer; or, the Whole Art of Measuring…*, printed for E. Bell et al., 1721. The page of the advertisement and the preface are unnumbered. Hawney mentions the sliding rule on pp. 327–328 and p. 334 in the section on gauging.
16. For information about Thomas Keith, see "Memoir of Thomas Keith, Esq.," in T. Keith, improved by S. Maynard, *An Introduction to the Theory and Practice of Plane and Spherical Trigonometry…*, Longman, Orme, Brown, Green, and Longmans, 1839, pp. xvii–xx.
17. W. Hawney, *Hawney's Complete Measurer…*, rev. ed., T. Keith, ed., for J. Johnson et al., 1798. The section on Gunter's scale is pp. 48–50, on the carpenter's rule is pp. 53–57. Pages 266–296 describe gauging, using the gauger's rule throughout. A mention of Jones is on p. 266.
18. W. Hawney, *Hawney's Complete Measurer, or, The Whole Art of Measuring…*, rev. ed., Thomas Keith, ed., M. Carey, 1807.
19. W. Hawney, *Hawney's Complete Measurer…*, revised by T. Keith and then by J.D. Craig, F. Lucas, Jr. and Neal, Wills, and Cole, 1813. The 1820 Baltimore edition was published by F. Lucas, Jr.
20. J. Thompson, *A Complete Treatise on the Mensuration of Timber*, Wright, Wilbur, & Stockwell, 1805, p. 99. Little is known about Thompson. He did write *The American Tutor's Guide: Being a Compendium of Arithmetic: in Six Parts*, E. & E. Hosford, 1808, which did not mention the sliding rule.
21. C. Hutton, *The Compendious Measurer, Being a Brief, Yet Comprehensive, Treatise on Mensuration and Practical Geometry. With an Introduction to Duodecimal Arithmetick…*, Hugh Maxwell, 1807, pp. 99–110, 163–175.
22. Karpinski, *A Bibliography of Mathematical Works Printed in America through* 1850, pp. 189–190. The oft-reprinted book was J. Bonnycastle, *An Introduction to Mensuration and Practical Geometry…. To Which is Added, an Appendix, Containing a Concise System of Gauging*, Kimber and Conrad, 1812. This was taken from the 10th London edition of Bonnycastle. The text discusses the carpenter's rule on pp. 208–212 and uses it in examples in the section on timber measure that follows. It discusses the gauging rule on pp. 243–247 of the Appendix and uses it in examples through p. 259. Other American printings included those of 1818 (2nd American ed.), 1823 (3rd), 1827 (4th), 1829 (5th), 1831 (5th), 1834 (copyright 1833 and 1834 by James Ryan, most subsequent printings were of Ryan's edition), 1838, 1840, 1842, 1844, 1846, 1849, 1853, and 1860. A description of the carpenter's rule is on pp. 203–206 of the 1834 edition.
23. J. Bonnycastle, *An Introduction to Mensuration and Practical Geometry…*, for F. C. & J. Rivington, 1823, pp. 269–279.
24. For a survey of steam engines in the US in 1838, see "Letter from the Secretary of the Treasury, Transmitting… Information in Relation to Steam-Engines, &c.," US House Document 21, 25th Congress, 3rd session, 1838. The results of this survey are discussed at some length in C.W. Pursell, Jr., *Early Stationary Steam Engines in America*, Smithsonian Institution Press, 1969, especially pp. 72–93.
25. T. Wyman, "Slide Rules of the Stanley Rule & Level Company and Other American Makers," *The Chronicle of the Early American Industries Assoc.*, vol. 54, no. 3, Sept. 2001, pp. 114–117.
26. J. Routledge, *Instructions for the Engineer's Improved Sliding Rule…*, George Piper, 1828.
27. J. Routledge, *Instructions for the Engineer's Improved Sliding Rule…*, Belcher & Brothers, 1844.
28. On Middlebury in the mid-nineteenth century, see S. Swift, *History of the Town of Middlebury in the County of Addison, Vermont…*, A. H. Copeland, 1859.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

29. T.H. McLeod, *Instrumental Calculation or a Treatise on the Sliding Rule*, George Smith, 1846. For biographical information about McLeod, I am grateful to Suzanne Douglas of the Sheldon Museum in Middlebury, Vermont. Other books that discussed only the slide rule were Ohio engineer J.L. Cordingley's *Instructions for the Sliding Rule, with a Description of the Several Lines upon It…*, Amos Derrough, 1841; W. Moulton, *A Treatise on Gunter's Sliding Rule: Designed as a Pocket Companion for the Mechanic, Manager, and Artisan…*, B.B. Mussey, 1843; and F.A. Bryant, *A Key to the Slide Rule: Embracing Most of the Mathematics Applicable to the Mechanic Arts; Together with the Art of Calculation in a Cotton Mill…*, printed by H. Chickering, 1848.
30. For biographical information about Hackley, see *The Univ. Quarterly*, vol. 3, Apr. 1861, p. 388.
31. W. Templeton, *The Operative Mechanic's Workshop Companion and the Scientific Gentleman's Practical Assistant…*, John Whale, 1845, pp. 69–76. Templeton (1796–1863), a native of Ayshire in Scotland, had been an engineer in the Royal Navy, spent some time in Java and Australia, and then returned to Britain, where he wrote a variety of technical works. He died destitute. See "London Association of Foreman Engineers," *The Mechanics' Magazine*, vol. 10, 11 Dec. 1863, p. 858.
32. C. Haslett, *The Mechanic's Machinists, and Engineers Practical Book of Reference…*, C.W. Hackley, ed., W.A. Townsend, 1859, pp. 42–47. Hackley mentions the novelty of some aspects of the book in the preface on pp. v–vii.
33. The first engineering school in the US was the US Military Academy at West Point, officially founded in 1802. The Rensselaer School, established in 1824, became the Rensselaer Institute in 1833, and granted its first engineering degree in 1835. Norwich University, a private military school in Vermont, was authorized to grant degrees in civil engineering in 1834. See T.K. Jewell, F.E. Griggs, Jr., and S.J. Ressler, "Early Engineering Education in the United States Prior to 1850," Int'l Eng. History and Heritage, *Proc. 3rd Nat'l Congress on Civil Eng. History and Heritage*, from the 2001 Annual Civil Engineering Conf., 2001, pp. 335–353.
34. A. Eaton, *Prodromus of a Practical Treatise on the Mathematical Arts: Containing Directions for Surveying and Engineering*, Elias Gates, printed by Tuttle, Belcher, and Burton, 1838. Both quotations are from p. iv of this book. The discussion of the slide rule is on pp. 187–188.
35. A. Palmer, *A Key to the Endless, Self-Computing Scale, Showing Its Application to the Different Rules of Arithmetic, &c*, Smith & Palmer, 1844. An 1842 edition of this work was published.
36. J.E. Fuller, *Telegraphic Computer…*, for the proprietor, 1852.
37. J.E. Fuller and A. Palmer, *Fuller's Computing Telegraph: The Great Master Key Rule by Which All Problems are Instantly Solved*, John E. Fuller, 1852.
38. For further information about Palmer and Fuller's instrument, see F. Cajori, *Aaron Palmer's Computing Scale*, and *John E. Fuller's Circular Slide Rules*, Colorado College Publication, 1909, pp. 111–119 and 120–121. See also B. Feazel, "Palmer's Computing Scale," *J. Oughtred Soc.*, vol. 3, no. 1, 1994, pp. 9–17; and B. Feazel, "Palmer's Computing Scale—Revisited," *J. Oughtred Soc.*, vol. 4, no. 1, 1995, pp. 5–8.
39. D.M. Knapen, *The Mechanic's Assistant: A Thorough Practical Treatise on Mensuration and the Sliding Rule*, Appleton, 1850. The quotations are from pp. 5 and 6.
40. Author unknown, *Instructions for the Engineer's Improved Sliding Rule, with a Description of the Several Lines Upon it, and Directions How to Find Any Number Thereon: Together with the Application of those lines to Multiplication, Division, the Rule of Three, &C., &C., The Mensuration of Superfices and Solids are Likewise Made Perfectly Easy, It is also Particularly Useful in Weighing all Kinds of Metals and Other Bodies*, Hermon Chapin, 1858; P.E. Stanley, "Carpenters' and Engineers' Slide Rules: Routledge's Rule," *Chronicle of the Early Am. Industries Assoc.*, vol. 37, no. 2, June 1984, p. 27.
41. A. Jillson, *A Treatise on Instrumental Arithmetic; or, Utility of the Slide Rule. Designed as a Pocket Companion for the Mechanic, Manager, and Operative*, Hartford, Case, Lockwood, and Brainard, 1866. I thank Pat Spencer for assisting me in finding information about Jillson's birthdate and occupation from census records.
42. *Manufacturer and Builder*, vol. 6, no. 9, Sept. 1874, p. 212.
43. See Kidwell, Ackerberg-Hastings, and Roberts, *Tools of American Mathematics Teaching, 1800–2000*, especially pp. 117–122.

Peggy Aldrich Kidwell is the curator of mathematics at the Smithsonian Institution's National Museum of American History. She tends the collection of astrolabes, slide rules, geometric models, computers, and related objects. Kidwell has a PhD in the history of science from Yale University, and she has published and done exhibits on aspects of mathematics teaching, mathematical instruments, and computing. Contact her at kidwellp@si.edu.

The Production and Interpretation of ARPANET Maps

Bradley Fidler and Morgan Currie
University of California, Los Angeles

A 20-year series of ARPANET maps produced by the firm Bolt Beranek and Newman (BBN) signifies the earliest efforts to represent an early and central piece of the modern Internet. Once a functional tool for engineers, they now serve as an aesthetic backdrop used without explicit recognition of their intended purpose.

An early map of the Advanced Research Projects Agency Network (ARPANET) appeared in September 1969 (Figure 1).[1] The map's origins are unclear, possibly a sketch drawn by someone at ARPA contractor Bolt, Beranek and Newman (BBN) or the University of California, Los Angeles (UCLA).[2] The diagram shows a circle tethered to a square by a single line, representing the connection between two machines: a time-shared computer system at UCLA, called a host, and an interface message processor (IMP).

The map outlines the basic topology of the ARPANET in its earliest form, a project that would soon be regarded as a massive success in computer networking, even though initially it was met with skepticism. At that time, computer networks were not entirely new, but the ARPANET introduced a general-purpose packet-switched network that connected heterogeneous, physically remote machines. According to its planners at ARPA, the primary benefit of participating in this experiment was resource sharing; rather than ARPA spending millions to duplicate computing resources across the country's "centers of excellence" that it funded, the ARPA Computer Network (as it was originally called) would enable researchers to access the resources of others remotely.[3] By the end of its life in 1989–1990, ARPANET technologies had served as a blueprint for a new generation of BBN networks for the private sector, the US Department of Defense, and the intelligence community, and more famously, it was a major source of the ideas, people, and technologies that led to the development of the modern Internet.[4]

Many subsequent maps would follow this initial sketch, in a 20-year series of maps produced by BBN, which published one or more a year throughout the network's life. The BBN maps signify some of the earliest efforts to represent the most significant antecedent of today's Internet. And while the ARPANET grew in scope and function during its 1969–1990 run—ultimately leading, by 1983, to it being a central component of the modern Internet—the official maps produced during its existence retained the same representational strategy. We have found no record of significantly alternate mapping strategies put forward by the ARPANET's active and technically sophisticated user community or by subsequent researchers, so the maps maintain over four decades of stability in the network's visual representation.[5]

We begin exploring the ARPANET maps' production by situating our work within critical cartography and science and technology studies (STS) to illustrate how the maps can be read for what they say about their material bases rather than acting as straightforward representations—what interpretive decisions led to their construction and influence the understanding of these artifacts after the fact? From there, we analyze the maps' historical origins. Specifically, we ask about the data-collection methods that preceded mapping, the design conventions used to visualize this data, and the parameterization work that determined what technical information the maps were intended to illuminate. Methodologically, this research includes analysis of primary source materials, such as the maps themselves. We also draw from discussions with those involved in the production of these maps and others working at BBN during the time of their use.[6] In our conclusion, we explore the significance of BBN's data-

 Published by the IEEE Computer Society 1058-6180/15/$31.00

collection practices and the maps' network graph form in terms of wider ARPANET representation and how these representations function alongside a more comprehensive historiography. Namely, we believe that the maps' form and its focus on the network subnet reinforce a historiography that also focuses on early years, eclipsing many of the significant technical and sociocultural changes that occurred as the network grew and changed.

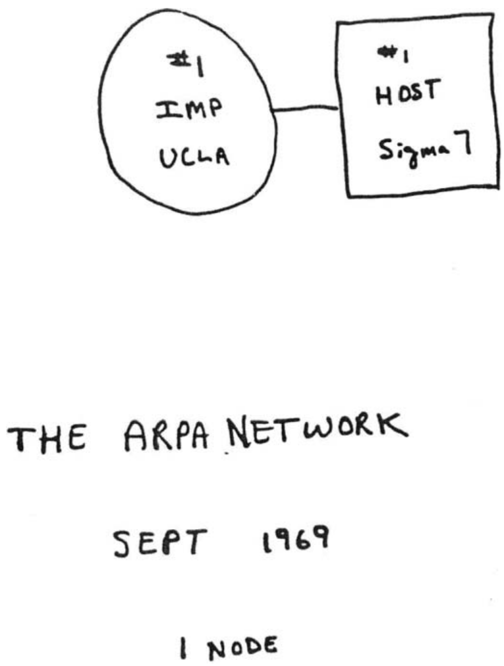


Figure 1. The first ARPANET node at the University of California, Los Angeles, September 1969.

Critical Map Reading

What does a critical reading of ARPANET maps entail? First, we use the term *maps* deliberately, drawing from cartographers J.B. Harley and David Woodward, who describe them as "graphic representations that facilitate a spatial understanding of things, concepts, conditions, processes, or events in the human world."[7] Graphic representations such as maps are composed of what Johanna Drucker calls "capta," or organized and parameterized constructions in graphical expression.[8] (Drucker avoids using the term *data,* shrouded as it is in the common misperception that data are equivalent to phenomena in the world.) That is, graphic visualizations are always based on the selection of measurement criterion, data, and visual metaphors—on actions that involve interpretation. We therefore set out to examine maps not as passive records capturing a priori data about the world but as constructions reflecting the choices of their designers and the conventions of their time. We use the term *critical* in the tradition of theoretical critique, with nods both to the Frankfurt School and Michel Foucault. Here, critique becomes a means both to excavate the maps' implicit knowledge claims—the information their creators decided constitutes "the" ARPANET, as revealed through data selection and design—and also to challenge the casual use of these documents as visualizations of the ARPANET as a whole. An abundance of literature in areas such as critical cartography, digital humanities, and STS present similar orientations.

Feminist geographer Sara McLafferty, for instance, argues that geographic maps present a detached, God's-eye view, representing the efforts of individuals or groups as dots and complex activity patterns as linear pathways.[9] The design of a map cannot capture the variegated spatial experience of everyday network activity. The ARPANET maps deploy a distributed, topological network graph form, an analytic design deriving in part from 18th and 19th century graph theory developed in the fields of mathematics, electricity, and chemistry. The design is one that likewise privileges a geometric, spatial perspective over relational or causal dimensions, often hiding the complex agencies and directional flows involved in technological projects. Digital humanist Phillip Gochenour also points out the potential of network graph models to reify political and economic structures in a way that makes conflict, difference, or malfunction invisible: only connection and smooth technical or social functioning can be represented in the graph; any broken or odd parts are left out.[10]

STS scholar Geoffrey Bowker has proposed a similar critical approach when examining classification systems. Classifications are information infrastructures constructed silently behind the scenes, operating transparently through common use. Yet when foregrounded—a method Bowker terms "infrastructural inversion"—the standard categories of a classification scheme become objects of historical examination.[11] Rather than telling us about the order of the world, their categories are contingent reflections of their time. Bowker and Leigh Starr argue that classifications have consequences both as a political force and as an organizing schema that shape social identities and technical possibilities. In no way neutral, classifications embody ethical and aesthetic decisions and myriad compromises. To study classifications, a researcher must ask what their schemas render visible and what they leave out, as well as how their use spreads. Classification research, in other words, provides a method with which to view the processes of classifications as an ongoing "crafting of treaties" and a lens through which to examine the ethical or social impacts of any classification schema.[12]

With our reading of ARPANET maps, then, we propose a similar tactic of inversion. We read the consistency of the map design as a choice selected from a growing body of available data. The continuities and systematic nature of BBN's depiction of the ARPANET allow us to read the maps as proposing a certain perspective of the network's operational years that may affect retrospective histories. Namely, the maps parameterize the ARPANET based on a crucial, but limited, set of technologies, as we explain below. Similar to a classification scheme, the maps' parameters as visualized in the network graph form will exclude what is not deemed to fit, and this exclusion can have subtle consequences. In this article, we focus on the context of the maps' production at BBN, leaving a closer look at the maps' ellipses—the information they make invisible—for a forthcoming paper on the topic.

Such a critical reading is salient because these maps have come to circulate as straightforward representations in the literature on the ARPANET. In *Mapping Cyberspace,* for instance, Martin Dodge and Rob Kitchin reproduce six maps over the course of six years to exemplify how the ARPANET grew from four to over 50 nodes.[13] Peter Salus's *Casting the Net* and Arthur Norberg and Judy O'Neill's *Transforming Computer Technology* also use these maps to illustrate the network's scope,[14] while Martin Campbell-Kelly and William Aspray's *Computer: A History of the Information Machine* deploys the maps to depict the ARPANET as a whole.[15] In *Inventing the Internet,* the maps illustrate the inclusion of DARPA's "centers of excellence" in the early network design.[16] DARPA itself features the map as its main representation of the network on its history section webpage, as does the ARPANET Wikipedia page.[17] A technical retrospective of the ARPANET and other DARPA projects, funded by DARPA and completed by the Institute for Defense Analysis, also uses these maps to illustrate the network.[18] The maps have also circulated as Web ephemera in social media posts, showing up as a constant Twitter fixture and in online journalism.[19] In all these examples, the maps are presented as straightforward signs, with little qualifications about their limited representational aims, or as an aesthetic backdrop for descriptions of the early ARPANET of 1969–1975, but not its later, more ubiquitous (and perhaps more influential) form. As we will explain here, characterizing the ARPANET's history based on this particular view introduces problems if it diverts historiography to a set of concerns from this earlier time period.

BBN's engineers designed the maps for specific purposes, never to represent the ARPANET as a whole. The maps were highly effective in fulfilling their role during the ARPANET's life and circulated among ARPANET engineers and users who surely understood, through experience, the difference between these maps and the broader ARPANET infrastructure, as well as the network's social complexities. Now, these representations should be viewed as partial constructs, especially for observers without experience using the network. The ARPANET map designers made the task of defining and revealing the network seem like a simple representational matter, and a noncritical reproduction of the maps does not complicate this assumption.

Map Design and Production

BBN, the Cambridge-based ARPA contractor that built, maintained, and ran the ARPANET, began its work in 1968 after it won the ARPA contract to develop the network; its initial charge was to make operational the network's basic functions of moving data from computer to computer. On Labor Day weekend in 1969, BBN delivered UCLA's IMP and connected it to the SDS Sigma 7 host.[20] The IMP acted as a minicomputer, slightly narrower and taller than a consumer refrigerator, and was engineered to serve as the host's link to the future network of other IMPs and hosts. The physical infrastructure depicted in the single-node map in Figure 1 was now in place. Within the same year, other computing centers joined the UCLA node, including the Stanford Research Institute, University of California, Santa Barbara, and the University of Utah.[21] These first four nodes appear on an increasingly iconic four-node map of similarly obscure origins as the single node map

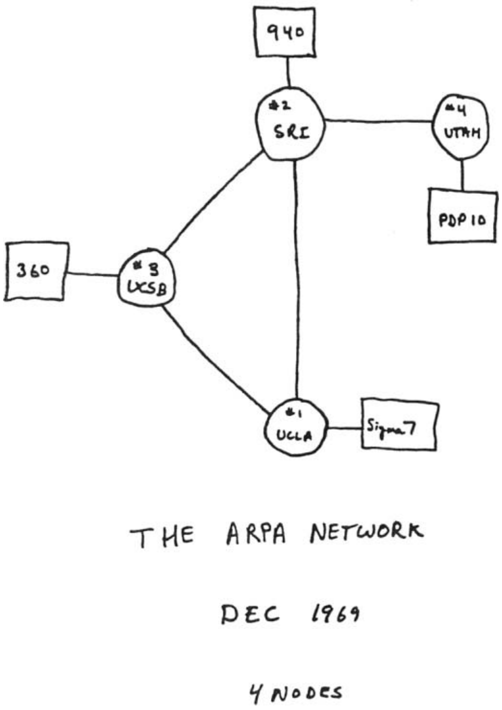


Figure 2. The first four nodes of the ARPANET, December 1969.

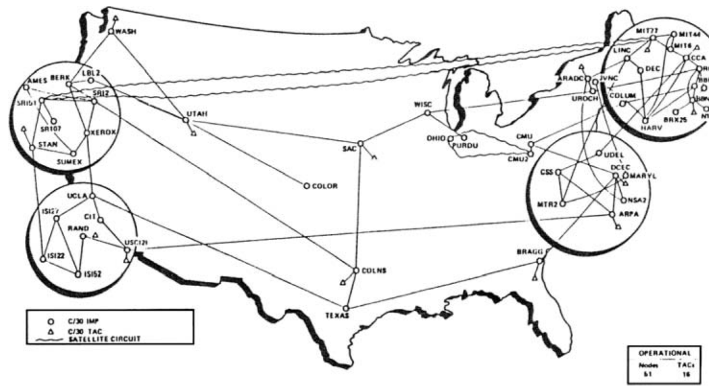


Figure 3. ARPANET geographic map, 30 April 1988.

(Figure 2); it serves as a simple schema of the network formed by the first four sites. BBN joined as the fifth node in January 1970. By March 1972, the network had 25 nodes.[22]

One of BBN's early tasks, a massive one, involved establishing a working subnetwork and maintaining and increasing its reliability as the ARPANET grew in size and complexity.[23] The subnetwork—or subnet—included all the IMPs and links that interconnected them, comprising the core physical infrastructure responsible for transporting data between the hosts. Building a working subnet entailed more than simply putting hardware into place. To ensure that packets of data would flow successfully through the IMP network, BBN developed custom IMP hardware and software, including the crucial routing algorithm. Importantly, the subnet had to be invisible to users, shuttling data between hosts automatically.

Bob Brooks, who managed the creation of BBN's ARPANET maps, locates their production in this early stage of the network's development.[24] It's perhaps not surprising, then, that the maps largely reflect BBN's responsibility at that time for the ARPANET subnet. This focus is evident in both of the two types of network graphs or maps BBN used to characterize this landscape, even though they visualize the subnet in slightly different ways. On geographic maps, set against an outline of the continental United States, the subnet's representation was limited to IMPs and the links connecting them (Figure 3). Logical maps, sometimes referred to as topological maps, represented the ARPANET sites much as a schematic subway map does subway stops: the connections between sites are maintained, although distances and relative locations are not proportional. Logical maps, however, also revealed each IMP's connection to a host, integrated within a representational scheme ordered by the IMP subnet (Figure 4). Yet ultimately the form taken by both maps relies on a small number of main components forming, and related to, the subnet: IMPs, hosts, links between IMPs and hosts, and links between IMPs (of these, only the hosts are not the subnet proper). Minor adjustments were made to represent certain changes to core technologies and to accommodate the growing number of nodes. To this end, the logical maps' underwent a stylistic update during 1974, displayed here in a 1977 map (Figure 5), yet the fundamental design of both map types remained focused on the subnet throughout the network's life, from the first months the network became operational until 1989, when its shutdown was nearing completion.[25]

Although the maps were likely originally produced on an ad hoc basis,[26] their use quickly spread as a medium that was "simple to understand, easy to distribute, and served a multitude of purposes," according to Elizabeth Feinler, who was familiar with the maps as director of the Network Information Systems Center (NIC) at SRI.[27] By 1975, both logical and geographic maps were released in official publications, such as the ARPANET Information Brochure and the ARPANET

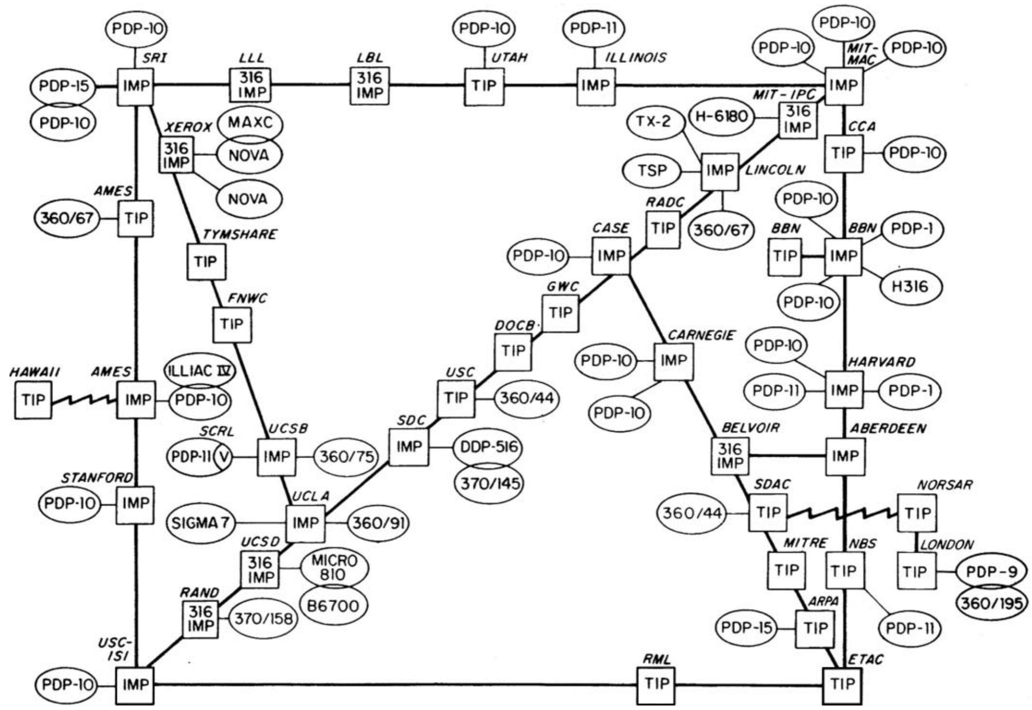


Figure 4. ARPANET logical map, September 1973.

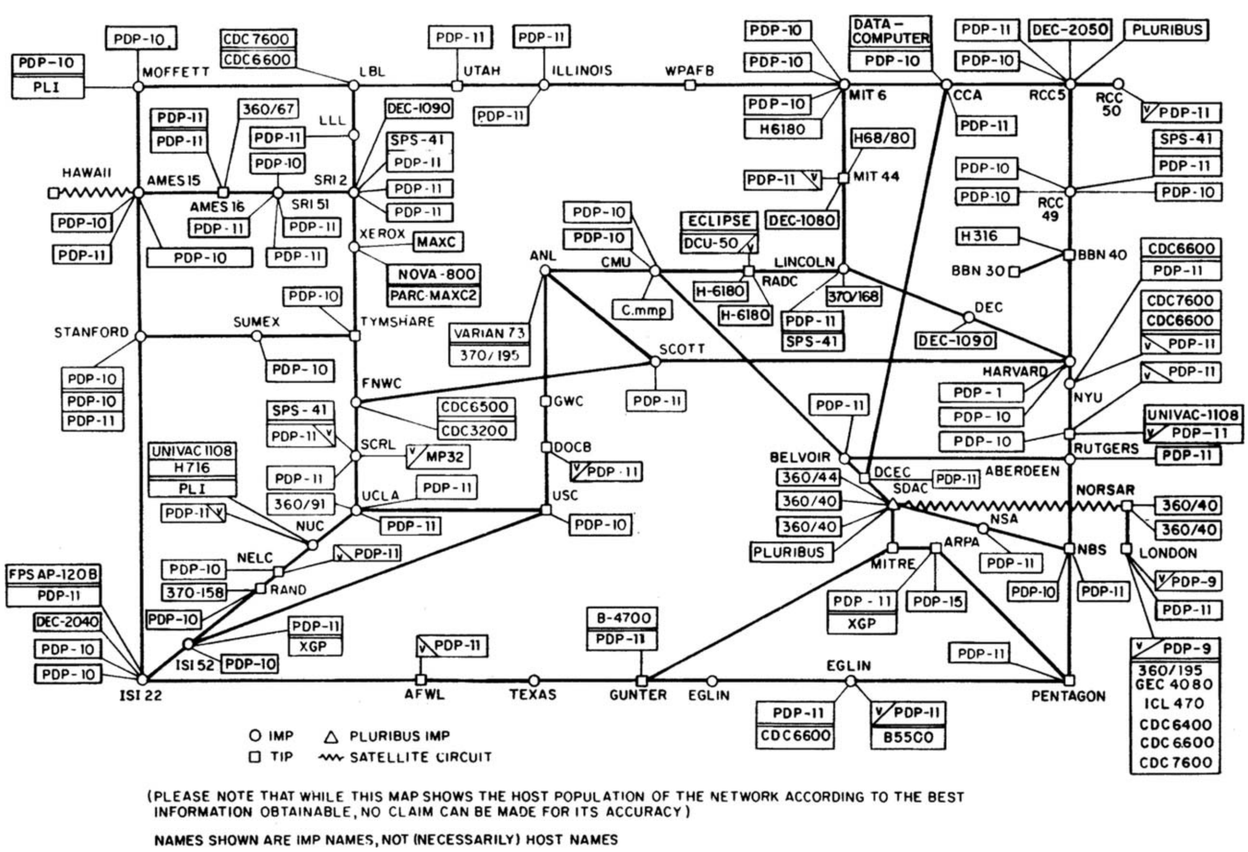


Figure 5. ARPANET logical map, March 1974.

Resources Handbook, documents that provided a range of information about the ARPANET for the user community.[28]

In sum, the mapping practices emerged during the earliest and most informal stage of BBN's management of the ARPANET. Especially in the early years of the network, implementing and improving the subnet—and keeping it functioning on a day-to-day basis—were a major priority for the BBN staff responsible for developing the IMPs.[29] The maps' network graph design represents this focus, as does the early data collection and selection practices. BBN retained the maps' design throughout their lifetime, even during the course of major changes as the ARPANET itself grew in complexity and as BBN became more sophisticated in its data collection and processing.

A Specific Parameterization

The maps' focus on the subnet stayed in place while the ARPANET grew in complexity with the introduction of new applications such as email and FTP, as it began connecting to both external and local networks, and as it reconfigured its institutional governance and access control policies (to name just a few developments). BBN had no reason to alter its interpretive strategies as staff could access more encompassing data that reflected these changes. Rather, BBN parameterized its maps by selecting from a larger set of static data it maintained on the configuration of the network, such as IMPs and their interconnections (the subnet), the type of IMP connection to its hosts, the name of each host, as well as line and modem numbers.

The maps themselves focused on three of these types of data, all related to the subnet. First, IMPs, as explained earlier, comprise the nodes around which the maps are structured. Interconnected IMPs are the ARPANET's backbone: repurposed minicomputers linked together to form the subnet and to ensure that a host's message travels to the destination host through the best available path of IMPs that lie between them. The maps reveal detail about IMP types, such as the terminal IMP (TIP), which allowed users direct terminal access to the network, sidestepping the need to go through a host machine, and newer IMPs, with more processing power and memory, such as Pluribus and C/30 IMPs. This subnet infrastructure, however, was invisible by design to those using the ARPANET. (Users would not, for example, need to know the path their data would take to reach a destination host.)

Second, hosts, found on the logical maps, were a common point of access to the ARPANET; by logging on to a host, users could access the network to connect to other hosts across the network. Hosts sent data to intermediating IMPs, indicating the distant receiver host to which a packet would be delivered. The maps reflect how hosts' relations to IMPs changed gradually from their original configuration in the early years of the network, when access was often through terminals in the same room as the host, to 1970, when the number of hosts per IMP began to increase and spread further from IMPs. For instance, the distant interface allowed up to 2,000 feet and appears on the maps as a D; the very distant host interface permitted arbitrary length and appears as VDHI.[30] Entire local networks of hosts that were increasingly attached to one IMP, such as the Stanford University Network, also show up on the maps but only as one entity, SU-NET.[31] Other specialized host functions appear as private line interfaces (PLIs) beginning in 1976 and terminal access controllers (TACs) in 1982. BBN labeled these different connections on the maps because the kind of interface, whether standard and distant host-IMP interfaces, would impact the NCC's diagnostic practices if a malfunctioning host was interfering with an IMP. In other words, these details were significant to the subnet,[32] while functions of the hosts that indicate ARPANET's infrastructure beyond the subnet were not represented.

Finally, the links on the map represent leased lines from telephone carriers, the connections between the subnet. The geographic maps appear to privilege geography to show these connections, although geography is pushed aside in the case of nodes in Hawaii or London, which simply show up as "outside the continental United States"; concentrations of nodes, power centers of the US network, are magnified to fit the nodes on the printed map. On later geographic maps, satellite connections were represented as uneven links (and experimental satellite connections were not shown at all.[33] In the case of both maps, all links between IMPs, even on geographic maps, are displayed logically, only revealing their origins and destinations: the actual geographic route of the ARPANET's leased lines, and the connections and transfers across the line's routes, were unknown to BBN. Indeed, even though the maps show complete, static links, a 1980 paper on the NCC's Network Utilities (NU) described the static network state, wherein all components are working and properly connected, as only an "ideal"—in reality, network components may, at any given time, have been malfunctioning or inoperative.[34]

All told, the maps focused on the ARPANET subnet, as they were structured around the IMPs and the IMP-to-IMP links that connect them. The maps do not, nor were they ever intended to, describe the broader ARPANET of users interacting with their host machines in an increasing number of ways nor, for example, the social and political hierarchies between them.

BBN's Wealth of Data

We also want to point out that the maps' focus on the subnet was based on conscious choices by BBN engineers, especially given that BBN increasingly had more access to different types of data. Moreover, BBN's data collection became more automated and sophisticated throughout the 1970s and in 1980, yet over the course of two decades, the maps' representational method and mode of production remained a job of manual selection among static data collected by BBN.

Significantly, the static map data described in the section earlier was not the only data collected by BBN. Throughout the ARPANET's life, BBN monitored and recorded two other distinct categories of information about the network. The second type of information collected was data on dynamic, temporary changes in the network, traffic flows, and errors such as line or IMP outages. A third type of data collected by BBN consisted of contact information of people responsible for each node as well as the telephone company responsible for the lines connecting them. The maps portrayed a selection only of the static data.

BBN also continued to select the static data put on its maps manually, even as data collection gradually became systematized and automated thanks to the pragmatic requirements of running the network. At first, BBN monitored information without any electronic link to the network, instead gathering that information ad hoc with calls to the individual nodes and based on reports from institutions and contractors that were installing IMPs,[35] but a series of incremental adjustments improved BBN's technical abilities to monitor the network. The first occurred when the firm joined the ARPANET as the fifth node, connecting its own IMP in January 1970. Through its IMP, BBN received human-readable status reports from each IMP on the network, alerting BBN staff to any errors. Then, BBN engineer Alex McKenzie led further efforts to develop new monitoring techniques in response to a growing demand for the network's reliability as the ARPANET grew,[36] creating the Network Control Center. The NCC allowed BBN engineers to gain knowledge of the networks' topology increasingly through direct access to the network's infrastructure.[37] In 1971, monitoring techniques improved again, as an NCC host compiled the IMP status messages for the staff, and expanded to include more information that helped BBN understand the actual state of the network.[38]

By 1980, the NCC used a UNIX NU program, which became the sole monitoring and control system in 1983. The NU monitored the static network state, such as the IMPs, lines, and now, more information about hosts, but also dynamic changes to the network, such as outages and traffic rates in a manner that was far more integrated than before. NCC staff could now centralize the three types of network information collected and stored through different means in the early NIC into a single database.[39] A broader set of monitoring techniques, combined with a centralized database, also allowed BBN to generate customized reports, such as traffic between nodes over a week or month.[40] It could even visualize these custom status reports on CRT displays as well as create a dynamically updated map of the network or of any component on the network, in an array of formats.[41]

Yet even with these new capabilities and types of data, the original map formats, created with the practices and technologies of the early 1970s, remained the same. Indeed, during 1971, the NCC mounted a large steel board on the wall in the front of the room to aid in monitoring, to which it manually transferred network information. Here, staff plotted the static network configuration as well as temporary changes or difficulties.[42] The board displayed information the NCC staff felt most relevant, such as recent problems or experimental code loaded in particular IMPs, and the names of the people working on particular issues.[43] Once the steel board was installed, it provided the site from which the map information was retrieved (although, later, presumably the information derived from the NU program), before being transferred to the BBN art department for illustration.[44]

In a further step of refinement, beginning in late 1969 or early 1970, Bob Brooks became responsible for transferring a subset of information about the network, such as topological information and site names, from BBN ARPANET staff to the art department for final illustration. Laura Selvitella, a long-time illustrator for BBN, received sketches of the broad strokes of data to be included and then organized them in the topological layout form, making decisions that rendered the sketches into the final illustration.[45] Her designs added a graphic formality to the data, rendering them as authoritative or at least seemingly comprehensive documents to audiences with less familiarity with the network.

In sum, the original choices about the kind of information to visualize in the maps were a deliberate selection from an increasingly broad set of data available to and collected by BBN. Among this data, whether gathered manually, directly from IMPs, or eventually by the NU application, staff made choices about what was relevant for inclusion in the maps, choices that remained consistent over the life of the network. This design remained the case, although not for lack of data or different visualization techniques. Nonetheless, the configuration of the first available logical map of 1970 would remain for the duration of the network's life. Below (or above) the surface of these maps, then, lies a world of additional detail at hand: different traffic levels between sites, extremely popular hosts versus those that were barely used outside of their institution, the extent to which a node carried purely localized traffic, or the distribution of ARPANET access throughout an institution (for example, through distant terminals or a local network), to name just some areas. We're reminded again of what Bowker and Star call the continual crafting of treaties; the maps' narrow selection of data and network graph form emphasize an infrastructure that was central to a functional network, but they meant little to a user's everyday experience with that technology as it grew and shifted over time. A wealth of other concerns and technologies arose as the network expanded in size and differentiated beyond the subnet; here, we focus briefly on two.

First, the maps do not depict the different connections and densities of flow that diversified within the ARPANET. The maps do not visualize the network's widening sociotechnical infrastructure, one no longer composed of one host per IMP (as the maps originally denoted) but of hosts' diversifying user base and the interconnected networks of hosts, both locally and internationally. Durable patterns of highly local network use, for instance, emerged early in the ARPANET's history and remained at varying degrees in different places well into the early 1980s.[46] These practices, which were measured quantitatively by both the NCC and the UCLA Network Measurement Center, reflect local networks of multiple hosts connected to the ARPANET through a single IMP—connections that are not represented on the maps.[47] By the early 1980s, this trend expanded as local networks, such as SU-NET, and external networks, such as ALOHANET in Hawaii and the UK's University College of London, were connected to the ARPANET.[48,49]

Second, the maps' network graph model, in which all nodes appear equal, does not illustrate these hierarchies of control as ARPANET governance shifted. As the network developed, especially from 1975, access control regimes and security became major concerns. The major site of research into the properties of the network, UCLA's Network Measurement Center, was phased out, and the SRI NIC took on an increasingly influential role in implementing network policies and regulating access to the network. The rise of the SRI NIC related to the major shift in governance that occurred in 1975, when management of the ARPANET passed to the Defense Communications Agency (DCA), the defense agency that ran the network as an operational rather than experimental infrastructure.[50] This marked a time period that has been given little attention in the historiography of the ARPANET, when interest in controlling remote access to the network led to, among other things, the institutionalizing authentication of users in 1984.[51]

These are just some examples of major shifts in ARPANET use, technologies, and management that were not present in the first few years of the network's operation, from 1969 to 1970, when the maps were formalized. What is more, most ARPANET historiography parallels these maps by focusing on a structure of the network drawn from its early years, rather than the multifaceted phenomena that arose later.[52] The famous first four nodes are easily knowable, with the personalities of the people who operated them and specialized functions widely reported. But even though the ARPANET grew and

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

diversified, little has been reported in depth on the changing points of control and technological characteristics of the network. Reading these maps for the traces of these later uses, technologies, and management shifts that occurred in the ARPANET's final years highlights the need for further research.

Conclusions

As historically situated records, maps reveal much about the context of their production, including what they neglect to represent. As such, by reading maps critically, we can also imagine possible recuperations of the historical narratives that they've neglected. The case of the ARPANET maps is no different. We have demonstrated that BBN's priorities, data-collection practices, and mapping strategies were oriented around the ARPANET subnet, an early technological challenge of the network, and an infrastructure that was intentionally opaque to the hosts and, when functioning properly, invisible to users as well. There were other goals and priorities alive on the network at that time, but as argued, the subnet was at one point a central task and a major technological achievement of the network.

We believe that the map's network graph form reinforces the selections BBN made. The ARPANET subnet was designed to be hierarchically flat, at least in terms of its job to send packets to the right destination, since no IMP counted more than any other. This framework was explicitly conceptualized in the ARPANET's early design phase in 1968–1969: ARPA planners specified a flatly decentralized subnet, and BBN engineers made this possible. The ARPANET maps, which represent each node in similarly networked fashion, reinforce this description. Yet as Gouchenor argues, the network form also gives the impression of a smoothly functioning whole, and its white marginal space suggests completion. Rather than alluding to its narrow focus, the network graph form subtly reinforces a limited reading of the object it represents.

The subnet, a functionally nonhierarchical and homogenous infrastructure, with parts that operate equivalently, is therefore perfectly suited to the basic two-dimensional network graph form used in the maps' design. The network graph form does not easily accommodate other sociotechnical factors that operate on top of the subnet: the hierarchies of login access that developed among nodes or the directions of internal communication flows and flows from external sources. BBN's partial focus on the subnet suited a form that makes differences less evident. Yet as a result, the continuities and systematic nature in the maps' form, one so central to the subnet, encourage us to read them from a certain perspective based on the operational years, a view that may affect how retrospective histories depict the ARPANET's entire lifetime. Much of the literature that focuses on a time when the ARPANET was "generally non-hierarchical," "decentralized, collegial and informal,"[53] with "no central control point,"[54] neglects these later changes and the role that social ties and centers of privilege continued to have on its technical development. Although some authors are sensitive to the ARPANET's contradictory, heterogeneous parts,[55] what occurs in much of this writing is a conflation of early technical features with later social registers, not leaving room for the evaluation of lesser examined but vital points of control. It is the task of further historiography to excavate these features of the network.

Acknowledgments

Bradley Fidler thanks the ARPANET pioneers who informed and corrected the authors' understanding of how the ARPANET functioned and provided invaluable insight into the maps: Ben Barker, Bob Brooks, James Dempsey, Elizabeth Feinler, Jack Haverty, Alan Hill, Leonard Kleinrock, and Rick Schantz, with a special thanks to Alexander McKenzie and David Walden. The authors are responsible for any remaining errors.

References and Notes

1. V. Cerf and B. Kahn. "Selected ARPANET Maps," *Computer Communications Rev.*, vol. 20, 1990, pp. 81–110. In this paper, Cerf and Kahn selected 26 ARPANET maps of significance, including an index with a short note about the importance of each.
2. P. Salus to B. Fidler, email, Oct. 2013; Salus, who published these maps in *Casting the Net* (Addison-Wesley Professional, 1996), reports that one possible author is the late Jon Postel.
3. L. Roberts, "The ARPANET and Computer Networks," *Proc. ACM Conf. History of Personal Workstations*, 1986, pp. 51–58; doi:10.1145/12178.12182; S.J. Lukasik, "Why the Arpanet Was Built," *IEEE Annals History of Computing*, vol.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

33, no. 3, 2011, pp. 4–21; doi: 10.1109/MAHC.2010.11.

4. D. Walden and R. Nickerson, *A Culture of Innovation*, BBN, 2011, p. 110; http://walden-family.com/bbn/bbn-print2.pdf; A. McKenzie and D. Walden, "The ARPANET, the Defense Data Network, and the Internet," *Encyclopedia of Telecommunications*, Marcel Dekker, vol. 1, 1991, pp. 341–376.
5. This does not preclude two minor variations of the logical type map used internally at BBN since the early 1970s: one produced by a Calcomp map and plotting program and the other drawn by hand and included in a binder of site information used in the early 1970s. In addition, if there were other design standards for maps of electronic networks at this time, we are not aware of them, and they were not identified in discussions with BBN engineers.
6. We have identified with an endnote each sentence that is derived from a discussion.
7. J.B. Harley and D. Woodward, The History of Cartography, Volume 1: Cartography in Prehistoric, Ancient, and Medieval Europe and the Mediterranean, Univ. Chicago Press, 1987, p. xvi.
8. J. Drucker, "Humanities Approaches to Graphical Display," *Digital Humanities Quarterly*, vol. 5, no. 1, 2011; www.digitalhumanities.org/dhq/vol/5/1/000091/000091.html.
9. S. McLafferty, "Women and GIS: Geospatial Technologies and Feminist Geographies," *Carfographica*, vol. 40, no. 4, 2005, pp. 37–45.
10. P. Gochenour, "Nodalism," *Digital Humanities Quarterly*, vol. 5, no. 3, 2011; www.digitalhumanities.org/dhq/vol/5/3/000105/000105.html.
11. G.C. Bowker, *Science on the Run: Information Management and Industrial Geophysics at Schlumberger, 1920–1940*, MIT Press, 1994.
12. G.C. Bowker and L. Star, *Sorting Things Out: Classification and Its Consequences*, MIT Press, 1999, p. 7.
13. M. Dodge and R. Kitchin, *Mapping Cyberspace*, Routledge, 2000, p. 9.
14. A.L. Norberg, J.E. O'Neill, and K.J. Freedman, *Transforming Computer Technology: Information Processing for the Pentagon, 1962–1986*, Johns Hopkins Univ. Press, 1999, p. 175.
15. M. Campbell-Kelly et al., *Computer: A History of the Information Machine*, Sloan Technology Series, 2013.
16. J. Abbate, *Inventing the Internet*, MIT Press, 2000.
17. See www.darpa.mil/About/History/History.aspx and https://en.wikipedia.org/wiki/ARPANET.
18. S.G. Reed, R.H. Van Atta, and S.J. Deitchman, "DARPA Technical Accomplishments, An Historical Review of Selected DARPA Projects," vol. 1, Feb. 1990; www.darpa.mil/WorkArea/DownloadAsset.aspx?id=2678.
19. Twitter search for "Internet map 1969," http://boingboing.net/2013/07/04/arpanet-map-march-1977.html, http://gizmodo.com/5733671/this-is-the-internet-when-it-was-just-a-baby, and www.businessinsider.com/internet-in-1969-2014-1.
20. Other ARPANET sites such as SRI, UCSB, and Utah were freed of BBN's same responsibility and could devote time instead to devising and improving technologies to use the ARPANET more usable, the major example being the Host-Host Protocol, also referred to as the Network Control Program (NCP).
21. L. Kleinrock, "Creating a Mathematical Theory of Computer Networks," *Operations Research*, vol. 50, no. 1, 2002, pp. 125–131.
22. "*ARPANET Completion Report*," tech. report 4799, BBN, Jan. 1978.
23. J.M. McQuillan and D.C. Walden, "The ARPA Network Design Decisions," *Computer Networks*, vol. 1, no. 5, 1976, pp. 243–289; doi:10.1016/0376-5075(77)90014-9.
24. B. Brooks to B. Fidler, email, Oct. 2013. "At the time I was hired, Frank Heart's group was just installing the beginnings of the ARPA Network—ARPANET—3 nodes in California and one in Utah. For the first relevant Quarterly Technical Report to ARPA, I was asked to prepare simple maps of where these nodes were located geographically and how they were connected logically."
25. The "ARPANET Completion Report" (1978) includes a December 1969 geographical map and a December 1970 logical map; both series conclude in 1977, the year before the report was released.
26. B. Fidler and A. McKenzie, discussion, Oct. 2013.
27. B. Fidler and E. Feinler, discussion, Mar. 2014. The maps originally circulated according to the needs of a handful of interest groups and as evidence of progress to ARPA, in particular through BBN's quarterly technical reports (Fidler-McKenzie discussion, Oct. 2013); they also appeared in various BBN presentations and proposals (B. Brooks to B. Fidler, email, Oct. 2013). Both types of map are prominent in informational materials provided for the major 1972 demonstration at the International Conference on Computer Communication in Washington, DC, that introduced the ARPANET to the broader computing community ("Honeywell at Bolt, Beranek and Newman, Inc.," Oct. 1972 brochure produced by Honeywell Information Systems, courtesy of J. Haverty). Because logical

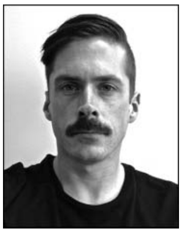
IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

maps added a further layer of detail by listing the hosts connected to the IMP subnet, they may have been an initial response to user requests for host information (Fidler-McKenzie discussion, Oct. 2013). The SRI Network Information Center (NIC) would also send out these maps to individual users who requested them for information about which sites were on the network (E. Feinler to B. Bidler, email, Aug. 2013 and Mar. 2014). They also appeared on BBN-produced t-shirts (B. Brooks to B. Fidler, email, Oct. 2013).

28. E. Feiner et al., "*ARPANET Resources Handbook*," tech. report NIC 45601, SRI Int'l ARPANET Network Information Center, 1978; ARPANET Information Brochure, Defense Communication Agency, 1978.
29. D. Walden with B. Fidler, discussion, Mar. 2014. Other staff at BBN were actively involved in the development of other areas of the ARPANET and related technologies, such as host-level protocols that made the network generally usable and advanced operating systems that became extremely popular on the network; Walden and Nickerson, *A Culture of Innovation*, chaps. "Data Networking at BBN," "Email," and "Distributed Communications"; www.cbi.umn.edu/hostedpublications/pdf/CultureInnovation_bbn.pdf.
30. "*Specification for the Interconnection of a host and an IMP*," BBN report 1822, May 1978. It is unclear how long VDHIs extended on the ARPANET.
31. This network is visible in the June 1983 logical map. Little is known about these local networks, and they remain an important focus for historical inquiry.
32. B. Fidler and A. McKenzie, discussion, Oct. 2013.
33. Satellite connections were used with satellite IMPs (SIMPs), which curiously do not show up on the maps. Satellite connections brought with them their own host of challenges.
34. P.J. Santos et al., "Architecture of a Network Monitoring, Control and Management System," *Proc. 5th Int'l Conf. Computer Communication*, 1980, pp. 831–836.
35. B. Fidler and M. Thrope, discussion, Feb. 2014 and Oct. 2013. A. McKenzie, "The ARPA Network Control Center," *Proc. 4th Data Communications Symp.*, 1975, pp. 5.1–5.6. BBN monitored unplanned events (such as outages) and often passed trouble calls to the homes of some of its employees.
36. B. Fidler and A. McKenzie, discussion, Oct. 2013. See also McKenzie, "The ARPA Network Control Center," pp. 5.1–5.6. Aside from the first node, the UCLA Network Measurement Center, other sites expected a stable and operational network. The NCC is sometimes referred to elsewhere as the Network Operations Center.
37. A detailed account of the early stages of the NCC, as well as a broader overview of its monitoring and control capabilities, can be found in A. McKenzie et al., "The Network Control Center for the ARPA Network," *Proc. 1st Int'l Conf. Computer Communication*, 1972, pp. 185–191.
38. McKenzie, "The ARPA Network Control Center," pp. 5.1–5.6. It appears that messages expanded again around 1974.
39. The 1978 inception date is listed in J.G. Herman's bio in S.L. Bernstein and J.G. Herman, "NU: A Network Monitoring, Control, and Management System," BBN, 1983, pp. 478–483. Although the details remain unclear, between the 1975 configuration and the 1980 implementation of NU, a TENEX-based "U program" may have been used in some aspect of network monitoring. J. Dempsey to B. Fidler, email, Oct. 2013.
40. Bernstein and Herman, "NU: A Network Monitoring, Control, and Management System," *Proc. IEEE Int'l Conf. Communications: Integrating Communication for World Progress* (ICC 83), vol. 1, June 1983, pp. 478–483. Thanks to James Dempsey for this source.
41. Santos et al., "Architecture of a Network Monitoring, Control and Management System," p. B5.2.6.
42. The large network map is also described in K. Hafner, *Where Wizards Stay Up Late: The Origins of the Internet*, Simon & Schuster, 1998, p. 168. Alex McKenzie corroborates this account.
43. J. Dempsey to B. Fidler, email, Oct. 2013.
44. B. Brooks to B. Fidler, email, Feb. 2014 and Oct. 2013. Bob Brooks had this role until 1986.
45. B. Fidler with Laura Selvitella, discussion, Feb. 2014.
46. A January 1993 BBN report of local traffic statistics is available in the William Naylor collection, KCIS Archives, UCLA Library Special Collections.
47. RFCs authored by A. McKenzie, starting at #378 (Aug. 1972) and ending at #612 (Dec. 1973); L. Kleinrock and W.E. Naylor, "On Measured Behavior of the ARPA Network," *Proc. Nat'l Computer Conf. and Exposition*, 1974, pp. 767–80; doi:10.1145/1500175.1500320.
48. See logical maps from 1983 and later. Accessed with permission from the Computer History Museum.
49. M. Schwartz and N. Abramson, "The Alohanet: Surfing for Wireless Data [History of Communications]," *IEEE Comm. Magazine*, vol. 47, no. 12, 2009, pp. 21–25; P.T. Kirstein, "The Early History of Packet Switching in the UK," *IEEE*

Comm. Magazine, vol. 47, no. 2, 2009, pp. 18–26.

50. The increasing role of the NIC in ARPANET operations and policy is illustrated in the DDN newsletter discussions; see www.rfc-editor.org/rfc/museum/ddn-news/ddn-news.n1.1, www.rfc-editor.org/rfc/museum/ddn-news/ddn-news.n1.1, www.rfc-editor.org/rfc/museum/ddn-news/ddn-news.n18.1, and www.rfc-editor.org/rfc/museum/ddn-news/ddn-news.n36.1.
51. Between 1973 and 1975, BBN developed a distributed system of access control and basic user metadata collection (see BBN reports 2869, 2976, and 3089, and R.E. Schantz, "BBN's Network Computing Software Infrastructure and Distributed Applications (1970-1990)," *IEEE Annals History of Computing*, vol. 28, no. 1, 2006, pp. 72–88.). The final 1984 implementation as TACACS is linked to these early efforts in Walden and Nickerson, *A Culture of Innovation*, p. 461.
52. A first generation of foundational works on the ARPANET emerged in the 1990s and, understandably, focused on the emergence and early operation of the network. See Hafner, *Where Wizards Stay Up Late*; Abbate, *Inventing the Internet.*
53. Abbate, Inventing the Internet, p. 54.
54. K.L. Hacker and J. van Dijk, *Digital Democracy: Issues of Theory and Practice*, Sage, 2000, p. 20.
55. See G. Downey for a nuanced portrait of the network's heterogeneous and homogenous parts in "Virtual Webs, Physical Technologies, and Hidden Workers: The Spaces of Labor in Information Internetworks," *Technology and Culture*, vol. 42, no. 2, 2001, pp. 209–235.



Bradley Fidler is an assistant researcher in the Department of Computer Science at UCLA, where he studies the social and economic dimensions of networked computing. Fidler has a PhD in history from UCLA. Contact him at fidler@ucla.edu.

Morgan Currie is a PhD candidate in the Department of Information Studies at UCLA. Her research focuses on the history of data practices in governance and current applications of open government data. Currie has an MS in new media from the University of Amsterdam and a masters in library and information studies from UCLA. Contact her at msmorgancurrie@ucla.edu.

"There Is No Saturation Point in Education": Inside IBM's Sales School, 1970s–1980s

James W. Cortada
University of Minnesota

IBM's Sales School has provided the company's formal sales training since 1914. Having attended as a student in the 1970s and worked as an instructor in the 1980s, James Cortada describes the training newly hired IBM salespeople received during this era.

The words "There is no saturation point in education" are etched at the entrance of what used to be IBM's education center in Endicott, New York. In that building, generations of IBM sales personnel, engineers, managers, executives, and customers took classes during the 20th century. My first class, an introduction to IBM in October 1974, was conducted in that building. It had been a constant belief of IBM's long-time head, Thomas Watson Sr., that all staff should receive continuing education in their work, and no more so than his salespeople and management. Between the 1930s and the end of the 1970s, probably every American IBM sales representative (as they were officially called) and sales manager (called marketing managers) came through that building. By the end of the 1970s, other education centers around the world were providing similar services. The centerpiece of that education was also the longest running class, called simply "Sales School." The first was taught by Watson himself in 1914, one of his first steps to professionalize the company's sales force and to instill in them, and the company at large, many of the values and practices he had learned and used earlier at NCR.[1]

Prior to 1914, there had not been any formal selling processes at IBM. As in other companies, selling had not evolved into a profession; rather, people learned on the job. However, in some large companies, attempts were beginning to be made to fix that problem, certainly by the 1890s. At NCR, sales personnel had been taught closing techniques and approaches that later surfaced at IBM.[2] In the 1970s and 1980s, my time with this training, IBM's competitors were not doing formal sales training; they only trained employees about products, and tended to hire experienced salespeople, including many who had worked at IBM. So, the IBM Sales School was unique.

This article is about my personal experiences with Sales School, first as a student and later as an instructor. I also describe the importance of that class within IBM's sales culture. There is only one brief account describing that very special class, written in the mid-1980s by an IBMer, "Buck" Rogers, who spent his entire career in sales.[3] However, when IBM salespeople write about or discuss their careers, they typically mention when they attended Sales School, because it was a milestone event. Only after successfully completing it was an individual allowed to sell to customers—to carry a quota. It was as much a rite of passage as it was specific training about selling and how to do sales work. Additionally, I briefly describe the "structured sales call" because it embodied a set of practices that IBMers found useful for the rest of their lives as a masterful way of communicating. For that, there are no histories, other than the few pages written by Rogers while describing Sales School.

Sales School in IBM History

Sales School varied over the years, but its principle purpose remained constant: first, to teach IBM's values, such as excellence in all that one did and the virtue of a customer-centric culture, and second, to instruct sales personnel on how to "manage" (think "service" or "control") their customer accounts, how to navigate the sales cycle that IBM's products

 Published by the IEEE Computer Society

required, and how to communicate (think "convince") customers to implement a salesperson's recommendations. Walter D. Jones, a salesman in the company in 1914 when Thomas Watson Sr. became general manager, reported years later that all salespeople were required to take a course on the company's products that same year and a general refresher course for "older salesmen."[4] These objectives were taught in many forms, from the one class in the 1920s through the 1950s, to multiple classes from the 1950s to the present, including in Sales School. Beginning in 1925, Watson Sr. and other senior executives used the annual meetings of the 100 Percent Clubs as occasions to instruct on the techniques and attitudes of a sales representative. Topics included having a plan for what to do and what to recommend to customers, collaborating with colleagues in creating and implementing plans, communicating (selling) with customers, tracking progress, and always displaying an optimistic confidence that ambitious targets will be met based on solid planning. In the parlance of sales representatives, much of what they learned in Sales School was about the "basics" of sales, of "blocking-and-tackling"—in other words, the routine details of their job.

By the 1930s, Sales School normally ran for a week, but by the end of the 1960s, it was more often two weeks. As a student in the 1970s and then as an instructor in the early 1980s, I went through the two-week version. By then, every sales division had one- or two-week Sales Schools. Mine was in the Data Processing Division (DPD), which was believed to have descended directly from the sessions taught by Watson Sr. decades earlier, long before DPD had been created, because it was designed for IBMers who sold to the company's largest customers. The various sales divisions over the years have had similar classes to teach selling communications and how to manage clients, but tailored to their needs. For example, typewriter sales representatives spent more time learning how to "pitch" their products than how to manage accounts, whereas DPD marketing representatives had to understand how to manage client relations, because that was their most important strategy for maintaining large revenue flows. In each division, cases and examples were based on products and circumstances relevant to those students. Specialized sales groups, such as the Federal Systems Division (FSD) that sold to the US Government, normally used experienced DPD sales representatives in combination with consultants and software experts, so fewer of these individuals actually went through Sales School.

Records on how many people attended an IBM Sales School have not survived. However, anecdotal evidence suggests that every salesperson between the 1940s and the mid-1990s (when rapid acquisitions of small consulting and software firms flooded the ranks of IBM's sales staff) went through the class in its variant forms, depending on which sales division they entered. That would mean more than 100,000 individuals probably experienced Sales School. I know that during the period of the mid-1960s through the 1980s, it was not uncommon for several thousand sales persons to join the company each year just in the United States, and roughly close to that same number across the rest of the world. IBM was relatively small before World War II, so the number going through this training normally would have been accommodated in one or a few classes per year, often scheduled as needed. Nearly a century later, in the early 2000s, it was not uncommon for IBM to hire more than 4,000 new salespeople worldwide per year.

Before proceeding further, I should say what Sales School was not. Sales School did not teach the specific features and functions of products, nor did it teach the specific terms and conditions of IBM's contracts. Knowledge about those items was acquired either in other courses or through the extensive mentoring and training programs in branch offices. During the period from the 1960s to the end of the century, approximately 75 percent of all entry-level training occurred in various forms in branch offices. Sales School was always reserved for teaching sales communications, sales strategy, and how to manage sales cycles with customers. Many of the techniques taught in the 1920s were still being taught in the 1980s, the subject of much of what I discuss in this memoir. The methods and subjects taught in the United States were the same as those taught to sales personnel all over the world. In other words, a Sales School taught in Europe had essentially the same agenda as an American one. Indeed, when I was in instructor we interacted with the other schools to coordinate curriculums.

By World War II, when the class had become a routine part of the company's educational infrastructure, the instructors were salespeople who had practiced their trade in an outstanding manner and held great

Figure 1. Medallion embedded in the pen and pencil sets presented to Sales School instructors as they moved to new assignments.

promise as future sales managers. "Outstanding" was defined as having achieved their targets (quotas) back-to-back for several years; by my time, four in a row was normally required. Such salespeople were regarded as some of the most proficient in their branch offices and within their sales divisions. They had won outstanding sales awards, had garnished high customer satisfaction as reported by customers to IBM managers and through surveys, and they had successfully overcome severe competitive crises. They were seen as having "strategic vision" and strong teaming skills and were both politically savvy and excellent communicators. They were well versed in the current issues of the market, customer relations, and the product line. To ensure all instruction stayed fresh, current, and relevant, instructors were rotated in and out of Sales School in 18- to 24- month tours. Thus, examples of situations used in lectures came out of events that had occurred recently in the life of each instructor.

Normally, the overwhelming majority of instructors later became first-line managers in sales branch offices leading teams of five to 15 salespeople. They came back into the branch office having honed personnel management skills because they had to deal with the myriad problems of individual students and had learned to lead, persuade, and communicate to subordinates. They also continued to sharpen their already fine selling skills by teaching and watching others in Sales School. They normally did well in their first management jobs, and many went on to highly successful careers, rising right into the senior ranks of the firm. By the time I got involved as an instructor, Sales School instructor alumni had formed an invisible club within the firm. If you had been a Sales School instructor, you were assumed to be an outstanding IBMer even before you opened your mouth. If managers or executives who had been instructors themselves were interviewing an IBMer who also had been an instructor there was an instant bond, a respect that proved advantageous.

And how did you know if someone was a member of that club? In addition to reading a person's resume, for decades when instructors left Sales School to take on new assignments, they were given pen and pencil sets. Although these varied in design over the years, they normally had a medallion embedded in them that was so distinct you could spot one from across the room (see Figure 1). Sales representatives had long been taught to scan the office of whomever they were calling on to see pictures of family and events or other ephemera that could be used to open a conversation about a shared interest. So, it was normal to spot that medallion and then open the conversation acknowledging the mutual experience. Thus, these pen and pencil sets were prominently displayed, even in the offices of senior executives who had been instructors. They were a source of pride.

The most important reason why Sales School had such a cherished position in the firm was that it worked. Sales personnel, although they routinely discounted or thought silly the notion of the "structured sales call," had so internalized its techniques that they did not even recognize that they practiced it. We knew that because for decades highly experienced senior salespeople (mostly men at the time) always served as guest instructors, taking two precious weeks out of working "the territory" to help coach sales trainees. You would see them illustrate how to communicate with a customer in the classroom using the techniques taught to them often two or three decades earlier, executing them flawlessly. Yet they would deny they had done that until we showed them video of what they had said and compared it with the techniques they saw staff instructors teach during their two weeks at Sales School. The instructors and students were always impressed with the polish these representatives brought to the program.

IBMers used the same methods learned at this school with each other for the rest of their careers. Today, I can spot ex-sales professionals outside of IBM—say, in a community activity—because they continue to apply these methods for managing a situation or communicating with a group. I still use them today, and I am no longer at IBM. The methods worked because they were based on empirical research begun in the 1920s by various academics (such as communications experts), current business management thinking, and the practical experience brought by other practitioners from across IBM and even by customers who routinely were guest speakers at these classes.[5]

My Attendance, 1975

It was quite chilly in Endicott in December 1975. Several dozen trainees had checked in at the IBM Homestead, where thousands upon thousands of IBMers had stayed over the years while attending classes in town. Sales School was to be our last hurdle after a grueling 14- to 18-month training program, before we were let loose on customers and would carry a quota. As with all schools at IBM at the time, you had to successfully complete earlier courses before qualifying to attend Sales School. By the time we got there, we had learned about IBM's product line, computer technology, and how the company worked. We had been on many sales calls with sales representatives and had already attended a "Selling Fundamentals" class in which we were exposed to the structured sales call. In this, our last class, we would be asked to make seven practice sales calls built on a customer case study that each of us had prepared. These were based on live situations in our respective branches. Failure to pass Sales School essentially ended one's term at IBM. You had to compete against other members of the class for ranking, which affected your appraisal, which came in January at IBM, right on the heels of a cold December in Endicott.

My fellow "trainees," as we were called, represented a diversity new to IBM. Until the early 1970s, sales personnel in the United States were white men, all college graduates, most with degrees in business, engineering, and the hard sciences. They came armed with university degrees from major private and state universities. When I went through, the class was a mixture of African American, white men, and a few women. I was, I suppose, the token Hispanic, although I doubt anybody knew that. The trainees' backgrounds were broad. I was a history major, but I recall a religion major and someone who studied English literature. They were bright, good looking, ambitious, and articulate. You had to wonder how you got into such a crowd. They also demonstrated an ability to drink beer every night in town. They were all in their early twenties. I was 28 years old, the oldest—actually quite old by the standards of the day. Most trainees came into the company right out of college, with the exception of the rare military veterans who were closer to me in age. All our instructors were white middle-class American men in their late twenties or early thirties. Between them and the trainees, it seemed all regions had been accounted for, as they came together from around the United States.

Practice sales calls involved role playing, in which instructors were customers and trainees were sales representatives. We replicated a typical sales cycle of uncovering a business opportunity, exploring it for details about problems and their sources, conceptualizing a solution, overcoming objections, and getting the order. Other students observed our efforts; then we critiqued each other on what went well and how one could improve. Instructors did the same. Because everyone came to class with a specific case study, these calls could be tailored to the peculiarities of their industries and customers. For example, the US government had laws regarding procurement practices that trainees needed to know how to obey if they were slotted to sell to federal agencies. Large accounts in the private sector had less formal procurement practices, but their decision criteria might be different; for example, it might place more emphasis on return on investment than on the just plain old "who is cheapest" criteria used by the public sector. We had to be prepared for all of these differences, and among all members of the staffs in the 1970s and 1980s, the mix of experiences was sufficient to deal with these in the sales calls made in Sales School.

We devoted mornings to three activities. First, at 8:00 a.m. an ancient retired IBMer would come into the classroom, sit at a piano, and accompany us as we, yes, sang IBM songs! They were still sung in the 1970s, as they had been since the 1920s. By the 1980s, we thought this practice was quaint, but in the 1970s it was a tradition. Next, instructors gave 45–60-minute lectures, much

Figure 2. "Selling Fundamentals" class, 1975. Half of the members attended Sales School with James Cortada.

as at a university, about myriad topics, ranging from how best to execute various elements of the structured sales call (more on that later), how to handle difficult account situations, and what sources within the company one could go to for assistance. Trainees were urged to always bring up problems in their sales territories so that colleagues and management could help solve them. Normally, once a week customers would come in to discuss issues they faced in computing and, always, what they expected of their IBM sales representatives. These were terrific give-and-take sessions. Third, managers and executives from other parts of IBM would discuss their corners of the world. Executives gave pep talks about how exciting a time this was to be working for IBM.[6] On the last day, a senior executive would give a similar commencement address. I do not recall who spoke to us.

Our evenings were spent either preparing for the next day's sales call or in groups working out the solution to some sales problem to be discussed the next morning. Meals at the Homestead were excellent, the building gorgeous. Watson's wife had furnished it beautifully with the largest Oriental rug I had ever seen, which is still in Endicott at the local IBM museum, and with one of my favorite paintings, an impressionist scene of New York, circa 1914. When the Homestead was eventually closed, and its furnishing dispersed, I was saddened, thinking it lost, but a few years later saw the painting hanging outside the office of the Chairman of Board, Louis Gerstner, in Armonk, New York! There was a bar at the bottom of the hill on which the Homestead was located, and with some discretion, we followed another long-standing IBM tradition of meeting in the evenings. We left Endicott feeling that we had joined a long line of successful IBM salespeople, equipped with skills, confidence, and a tradition of success, as we also apprehensively thought about what January would hold for us.

Before closing out Sales School, a class president was selected. That individual was normally the one everyone thought did the best in school. The fact that they had been selected class president followed these individuals all through their IBM careers. Once, while attending a 100 Percent Club event in the late 1980s, among the guests introduced to the thousands of assembled IBMers was the class president of Sales School 1941. (For many decades, only one class a year was held. By the time I went through, they were held roughly once a month or two every six weeks.) A few weeks after the class ended, a class photo would show up in the mail. Although I cannot find mine, Figure 2 shows my "Selling Fundamentals" class, attended by sales staff and systems engineers; half were also students in my Sales School. By the time our class picture from Sales School came in, I was knee-deep in my first sales territory, and the class seemed long ago, not just six weeks before.[7]

I Became an Instructor, 1981–1983

Imagine my shock when I was appointed an instructor at Sales School in the fall of 1981. I was thrilled, although my wife was six months pregnant with our second child, Julia. We had to move quickly from New Jersey to Poughkeepsie, New York. The staff consisted of 14 instructors and a Sales School manager. We taught in building 5 at the plant. With a bumper crop of DPD salespeople being hired, we were teaching this class roughly once every three to six weeks. We also taught a one-week "Selling Fundamentals" course. Our classes typically had five to six instructors each, plus one or two experienced guest instructors from the field. Classes of 20 to 30 trainees were typical. Whereas classes in the 1970s were roughly 20 percent women and had a few African Americans, by the 1980s, women comprised a third or more and African Americans had also increased their presence.

The format of Sales School paralleled what I had gone through myself: lectures in the morning, practice calls in the afternoon, tailored case studies, guest speakers, and so forth. The objectives were also the same, but the singing was gone. One reason why Sales School moved from Endicott to Poughkeepsie was to be closer to IBM executives located just south of us in Westchester County, where DPD and the corporation had their headquarters. It made it easier to schedule them as speakers. We could also pull in customers from the greater New York/Connecticut/New Jersey region.

The environment in Poughkeepsie was quite different from that in Endicott. Endicott felt more like the early days of IBM, the Watson era of the pre-1950s, whereas Poughkeepsie was ground zero for the company's large mainframe business. Its operating systems were written there, along with many utility and networking tools. The factory built all the System 370 mainframes sold in the United States and Canada. Surrounding towns housed ancillary facilities where chips were made (East Fishkill) or where these computers were tested before going out to customers (Kingston). Close to 10,000 IBMers worked in the area. My neighborhood was almost 100 percent employees, most of them scientists and engineers. I was the first person in sales they had ever met. I spent the next two years dissuading them of an incredible set of misinformed perceptions they held regarding what salespeople did. If you wanted to cash a check at the local supermarket, the clerks were not interested in seeing your driver's license for proof of identity (standard practice in the United States), rather they asked to see your IBM ID badge. The other large employers were the two nearby state penitentiaries.

My colleagues were a diverse crowd, reflecting the changing nature of IBM at the time. (There were three women, one African American, and the rest were Caucasian.) All were educated at American universities and, with the exception of one, had spent their IBM careers in the United States. As I recall, about six of us had wives who gave birth to children in 1982. It played havoc with scheduling instructors because none of our wives would deliver babies when they promised. With the exception of one, everybody wanted to be a marketing manager right away. When they achieved that wish, they all did well at it, moving on with their careers, although over time some migrated to jobs outside of IBM. I learned a great deal from them about the company, about selling skills, and about dealing with trainees. It seemed everyone brought special experiences that they were willing to share. We were a tightly knit team. Other Sales School instructors from prior years told us that this was normal. On occasion an instructor from an earlier time visited us, especially those who had worked in Endicott and now were located in White Plains, New York, about 90 minutes south at the national division headquarters for DPD.

Trainees were generally an impressive lot, with a good mix of gender and race. All were still young; indeed, they seemed very young to me. They had a diversity of undergraduate degrees, and many trainees now had advanced degrees, including MBAs from any school you could imagine. They also included a folk singer who wrote a song about Sales School, a member of an US Olympic track team, and an individual with a medical degree. As instructors, we kept our social distance from them, so we knew little about their political and social beliefs. We saw a sprinkling of married trainees. Most expressed a healthy interest in sports, such as running, basketball, and golf, but at school they had almost no time for any of those activities. They tended to be a serious lot, having had their college party culture rubbed out of them in earlier classes. On the Thursday of the second week, there was a class party, however, followed by graduation the next day. Instructors came to these parties, which were always held at local restaurants, but we discretely vanished around 9:00 p.m. to let the trainees do and say whatever they wanted. But before we left, they made it a point to roast the instructors and to recall events from their class.

Life as an instructor was pretty intense. We had to study the trainees' case studies every night for the three sales calls we role-played the next day and be prepared to provide thoughtful advice on how to improve their performance. We videotaped every call so we could point out specific things they did well or could have executed better. Lectures were also taped so instructors either new to the program or giving a new lecture could receive similar critiques from more experienced colleagues. This is when I learned not to adjust my regimental tie that I wore always with a starched white shirt and, of course, a dark suit, while lecturing. Taking off one's jacket while lecturing was a pretty edgy, casual, even counterculture move. Remember, we had to be role models of professionalism at all times.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Each instructor was responsible for mentoring and advising up to a half dozen trainees while they prepared their case studies before attending class, then while in Poughkeepsie, and then in writing up and communicating an assessment of their performance in Sales School. Each of us was also responsible for the subject area we lectured about, and for documenting and updating lecture notes in case another instructor had to deliver the lecture. We liaised with other parts of the entry-level training program in our specialty area so that what we taught was consistent with what trainees had heard in earlier classes. Any redundancies in material were, thus, provided intentionally to address ongoing learning difficulties faced by trainees or to reinforce practices, such as sales call fundamentals and skills in using financial tools. My two staff assignments were to provide financial analysis training and to recruit guest speakers from IBM's executive ranks.

That second staff assignment was quite a revelation. I typically had to recruit division-level directors, vice presidents, division presidents, group executives to whom these divisions reported, and corporate officers. I never had one say "no" to speaking. The bigger problem was coordinating calendars. I was amazed at how supportive they were, regardless of division, even from such far-away parts of IBM as research and manufacturing—and even executives in Europe. Executives who had to cancel a speaking engagement normally recruited their own replacement. For example, I had IBM's chairman, John Opel, scheduled as a commencement speaker. He had to cancel, and rather than have a staff person deliver the bad news, he called me personally to apologize and to tell me that he had asked IBM's European chairman, Jacques Maisonrouge, who would be in New York, to speak. A few executives were not so professional, but on the whole supporting Sales School was considered crucial. Maisonrouge met with our class in New York at 590 Madison Avenue, where he delivered a rousing commencement speech, answered all manner of questions from the trainees, sang IBM's anthem, led us in singing "Ever Onward," and reflected on both his sales career and his attendance at Sales School in Europe in the late 1940s. Fantastic!

We often learned new things from our speakers that the rest of the company had yet to hear. In 1983, for instance, we learned that someday the power of a mainframe would fit into a device the size of a shoe box and that it was already in the labs. A speaker from corporate strategy shared with one of our classes—for the first time I think with any group in IBM—a slide that eventually became one of the most famous used by IBM in the 1980s. It boasted that the company would be a $100 billion operation by 1990 with a line graph showing revenues going straight up every year. At the time, I thought this was ridiculous, naive, so I asked for and he gave me the slide, which I still have. I wanted to keep it as an example of hubris. But his talk was well received by the trainees.

There were difficult times too. I had a trainee from Ohio that tried to commit suicide by taking an overdose of sleeping pills because the pressures of Sales School were too much for her. She was rushed to a hospital in the middle of the night; I reached her mother, and made arrangements for an IBMer in Ohio to go to her home and take her to the airport. A colleague in New York picked her up at the LaGuardia airport and brought her directly to the hospital. Once in a while, trainees drank too much. I rescued one from the local jail for that indiscretion, kept the incident quiet within the staff, and never reported it to his manager. The trainee was only about 22–24 years old and looked like a teenager.

There was one period in my two years when we had so much work that we literally ran one or two classes per week for seven months. At the time IBM was aggressively hiring sales personnel but was not expanding the entry-level training staffs to accommodate them. For our little team, that meant meeting on Saturday mornings for two or three hours to debrief on what went well and what didn't in the just-concluded class and what we had to do to close it out. Then we met on Sunday afternoons to figure out what we needed to do for the incoming class, greet the next crop of guest sales instructors, and brief them on the class and what we expected of them. We spent our evenings preparing for our sales calls or lectures for the next day. It was mentally exhausting, not to mention difficult on our families, who did not understand why we had to maintain such insane hours for so many months.

However, we met more than a thousand future salespeople, and nearly a hundred speakers, some of whom would go on to help our individual careers based on our positive professional treatment of them. For at least 20 years after being an instructor, people would come up to me at company meetings and say something like "Jim, you were an instructor in my Sales School" or "You were a tough critic on one of my sales calls." The 14 of us instructors remained friends for years, long after some had left IBM.

Because we were located at a plant site, we took our classes on tours of the manufacturing facility, which always proved to be a fun experience. Sometimes we also took them over to the software laboratories. We brought in speakers who had been our customers in the field, which proved to be wonderful experiences for all of us. On occasion, if an important customer was visiting the plant to see his or her company's computer being built, we would hear about it and immediately try to get that individual to meet with our students. These people who ran IT operations in large American corporations somehow always seemed to find a way to make time for that.

We could also attend lectures given by IBM scientists and engineers and sit in on other customer briefing programs held in the same building. In the two years I was in Poughkeepsie, I made it a point to try and attend at least one briefing every week, as they were held downstairs in the same building as our classes. I heard about IBM's latest products, what customers were doing with computers, issues such as cost justification of IT, and the rising tide, at the time, of Japanese competitors. Because I had already published two books on IT management, I was occasionally invited to give talks to groups of customers on related topics. Events downstairs helped me to stay current on issues. I knew the day would soon come when I would be back in a branch office with people looking to me for guidance and knowledge.

From our first day, we instructors knew we would go out with a promotion to marketing manager, so were keen to get going on our destinies. After a year on staff, we began figuratively to look at our watches, letting our manager know how eager we were to get "promoted out." At 18 months, an instructor was highly skilled, knew the teaching job well, and could not wait to move on. Of course, this had to be done keeping in mind the arrival of replacement instructors and the best placement for the veteran. The latter was a difficult task because geographic preferences and one's earlier branch experiences were factors. In my case, I was willing to go anywhere in the United States, but because I had experience with manufacturing and process accounts, I had to be placed in a sales office with those kinds of customers. It would not work to put me in a branch office that sold to Wall Street security exchanges. We also had to compete for openings with staff market support people who, also skilled in sales, were employed by the regional headquarters in which an opening existed. In other words, we had to compete against the "in-house" candidate.

Our competitive advantage was the experience we acquired in dealing with the human issues of our trainees—in appraising and teaching them—and our exposure to the rest of the corporation. We knew whom to call when there was a problem. Nor were we shy about doing so. This is an important observation because at the time, the company culture was formal and hierarchical. One did not simply pick up the phone and call, for example, an executive for help. You had to prepare for that request and run it up your chain of command. Yet executives understood the importance of people in the field getting to them quickly. Sales School instructors had learned through their experience that executives and upper management were approachable and normally more than willing to help; many in the field never knew that.

I left Poughkeepsie in November 1983 for a marketing manager's job in the DPD branch office in Nashville, Tennessee.

Structured Sales Call

I have mentioned several times the centrality of the structured sales call in the education and skill set of all IBM sales personnel. It is a subject that has not received attention by historians of the company, but it is important to understand both in terms of its role in IBM behavior and as part of the sales profession in general. Described internally as the "logical selling process," it held that all conversations were intended to persuade one to do something, ranging from agreeing with your point of view all the way to spending millions of dollars on whatever you were selling. It also could apply to convincing your spouse about the kind of renovation to do on your home or what vacation to take. The conversation had six elements that played out in one form or another: getting attention, generating interest in a topic, identifying a need and qualifying it, constructing a recommendation, presenting the case for it, and finally, closing the deal, which is asking for agreement or "the order." To do all of that, one could use a conversational architecture called the "planned sales call." It was not until the

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Table 1. Sample IBM approaches for closing an order.

Closing type	Example statement
Assumptive close	"What color would you like your computer to come in?"
Pending event close	"If you want this by the end of the year, you are going to have to order this now. May I go ahead and take your order?"
Directive close	"If all this makes sense, then the following are the three steps you and I need to take. Agreed?"
Puppy dog close	"Take this puppy home and try her out for 30 days with your children. If they don't like her, return the puppy and I will reimburse you for her purchase price."
Alternative close	"Would you like this computer in blue or red?"
Direct close	"Will you marry me?"

end of the 1920s, however, that the structured sales call had been codified into a teachable subject.

Sales representatives were shown how to initiate a conversation, how to spend time reducing customers' nervous tensions, often through the use of open-ended questions that required a sentence or more to answer, as opposed to a "yes" or "no" response. Creating interest was always about identifying a solution's potential benefits for the customer: "If we could solve the xyz problem, what would the effect be on your budget?" Inexperienced salespeople often rushed through this conversation, which aimed to funnel from the customer's broad goals and objectives, to a discussion about priorities among these goals and objectives, to how much by when and possible strategies for execution, to the specifics of the customer's needs. That needs phase could easily consume half the conversation, and so salespeople were trained to patiently and methodically ask questions that navigated the conversation from goals to specifics. We also taught them to verify that they (IBMers) understood what the customer was thinking and saying. We wanted them to validate continued interest, which could change during the conversation, because such a probing discussion often became an intellectual journey for the customer as well. All of this had to be done in 20 to 30 minutes! Why? Because too often that might be all the time you were given, particularly with a senior executive, such as a chief financial officer being asked to sign off on a large expenditure for your hardware.

Most inexperienced salespeople would jump to a solution (for example, "buy my product") about a third of the way into the needs discussion. An experienced salesperson, however, would use all that was patiently learned in the first half of the conversation to develop a solution (notice, not necessarily a product) that met the needs of the customer, and then present it. Tens of thousands of IBM sales representatives learned about FARs—features, advantages, and reactions. In presenting a recommendation, they were drilled constantly on the need for FARs. Explain the features of the solution, the advantages to the customer of each feature, and patiently get the customer's reaction. Some features and perceived advantages customers would agree with, others they would not. It was essential to understand from the FARs where customers were on a proposal. Always too, FARs had to be tied back to the needs established earlier in the conversation.

Then, and only then, could you ask for the order. There were many ways to do that, such as using assumptive statements ("What color would you like your computer to be?" or "After we get this machine installed, …"), or summarizing benefits ("In order to reduce your expenses by 5 percent this year, do you agree we should order this software now?"). Young salespeople were afraid to ask because they might feel rejected by a "no," whereas an experienced salesperson knew that "the selling begins when the customer says no." Sales representatives were also taught that part of the close involved agreeing to a plan of action for what IBM and the customer would each do next by some agreed-upon date. Salespeople were also taught how to craft a realistic commitment. Case studies proved useful in identifying what these might be. Table 1 lists example closing techniques taught to sales representatives. There were nearly a dozen different closing

Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

techniques, but I could only remember a few of the more obvious ones.

Jokes were often made about some of these. For example, "What closing technique is in widest use but can only be exercised in the last three weeks of December?" Answer: "Begging." So much for the planned sales call! But a hidden truth is that it paralleled the months or years of activities that led up to a sale: identifying a need, understanding the customer's issues, shaping a solution, identifying potential benefits, gaining the customers' concurrence to these, overcoming remaining issues (for example, adding something to the next year's budget), and agreeing to implement. With IBMers running on a quarter-to-quarter, year-to-year cycle, one could expect a great deal of order closing going on at the end of a month, quarter, or year.

We paid particular attention to teaching salespeople how to handle objections to their ideas. This is where highly experienced salespeople shined. They never became defensive or took an objection as a personal criticism; rather, they viewed an objection as a legitimate concern that needed to be overcome—something hidden that needed to be pried out of the customer, such as a prejudice in doing business with IBM or with someone of a different ethnic background. Thus, understanding the objection completely was crucial. To make sure that happened, a sales representative was taught to ask probing questions because customers rarely state their objections up front. Paraphrasing back ensured that the IBMer understood correctly. Then, only after taking these initial steps did a sales representative provide a positive response that was accurate and reasonable. Then he or she would ask for a customer response to confirm that the objection had been overcome.

This process may sound stiff, but with a little practice, it proved a natural way of talking if you were trying to sell something. It was so much a discipline that in the 1980s sales staffs were often given little blue plastic cards summarizing each step that they could keep in their wallets to consult from time to time (see Figure 3). I must have kept mine for more than a decade until it fell apart.

Tied to the planned sales call was a similar architecture for presentations. IBM salespeople were taught how to stand up in a conference room with a slide presentation, or in front of a large roomful of people, to persuade them of their viewpoint. The same elements

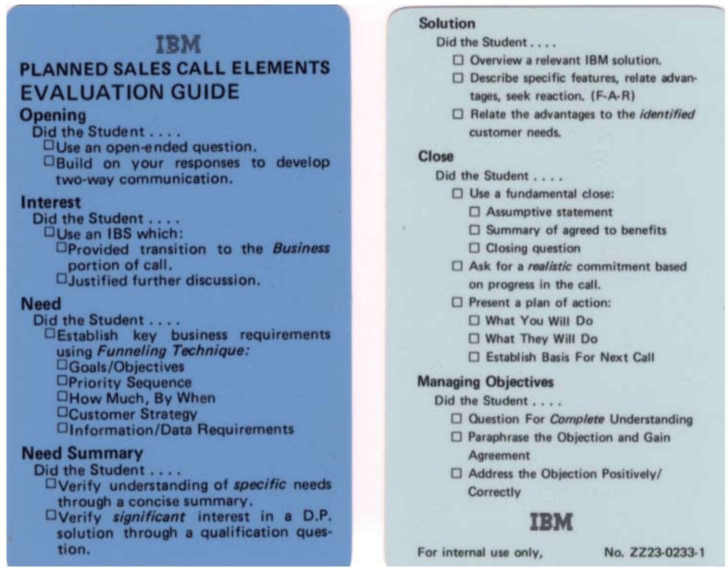


Figure 3. IBM sales training card. In the 1980s, salespeople at IBM were given plastic cards that listed the steps in a planned sales call and common customer objections.

I've described here appeared in some form or another in such presentations. One of my standard lectures in Sales School was on how to make presentations. Sales representatives would have seen enough of these to spot problems, so I would be introduced to a different class from the one I was working in as a highly experienced public speaker from New York there to talk about personal development. I would then make every mistake I could think of, and it would often take the audience close to 10 minutes to realize that it was all a spoof, at which point I would pivot into a properly done presentation on how to prepare and make presentations. All the usual tricks were in play: chewing gum, drinking a Coca Cola in a can, using yellow and red (the former so you could not see it, the latter to irritate the audience and be unreadable to the colorblind) or small fonts, reading prepared comments in a monotone voice, failing to make eye contact, not answering questions from the audience, and so forth. In Sales School, the seventh and last call trainees made was a closing presentation, which they had to prepare during the last week in class and present standing up. My job, of course, was to trot out all manner of objections to a trainee's proposal and, ultimately, only to agree if they had realistically convinced me of their idea. If not, they failed the call. As we told them, it was better to be treated roughly by us now than by real customers later. We were tough for that reason.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Sales School after I Left

The Sales School I described did not change significantly in the 1980s, whereas entry-level education as a whole evolved and over time became shorter. Although hiring declined in the 1990s, it did not end, nor did the need for education. By the early 2000s, attendance rose again. By 2009, Sales School was known as Global Sales School. Common themes remained similar to those of past decades—accelerating a seller's time to productivity, learning by doing, focusing on core and common skills—although the entire training program took months less than in earlier years. Experienced sales personnel coming into the company no longer had to attend Sales School, although inexperienced salespersons were still expected to do so. The planned sales call had a new name: Signature Selling Method (SSM). Trainees were still taught about critical sales activities, sell cycles, tools and assets they could apply to their sales efforts, and experienced teaming and collaboration. The director of Sales School in 2009, Paula Cushing, told me at the time that she had to cost justify the program, demonstrating its value to IBM in the increased productivity of trained personnel versus those that did not go through this and other entry-level training programs.[8] By then, it seemed that every organization in IBM had to justify its role in highly formal ways, including through economic justification and with formal PowerPoint presentations, surveys, testimonials, and employee feedback.

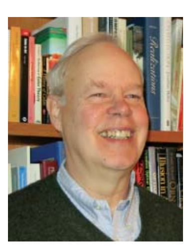
Some Final Thoughts

Of all the jobs I had at IBM, Sales School instructor was one of the most rewarding. Sales School was also the most pressure-filled course I think I ever took at IBM. That 40 years later I can remember the class suggests that it was transformative for me. It clearly was one of several activities that all salespeople underwent in their early years at IBM that reinforced the company's culture in its sales community. We learned a common language, a standard way to communicate with customers and our management, the fundamental sales cycle of working with customers, the firm's history, and a bit about the politics of a big corporation. When combined with the routines of a branch office, we could see how we became part of the selling ecosystem of which Sales School played an important role. Branch office milestones reinforced what happened with Sales School. These events included selling one's first computer system, attaining one's first 100 Percent Club, and receiving one's first Branch Manager's Award. As we traveled through the company over the decades, the common training—indeed attendance at the same class at the same time—had an ongoing bonding effect. As with the first classes taught by Watson Sr., we were forced to think about how we wanted to approach our customers, what we needed to learn, and what we wanted to say well. Sales School proved to be an important tool in making that possible.

I still practice its lessons, but hopefully, as an instinctive gesture, just like the 20- to 30-year sales veterans who did it so easily as guest instructors at Sales School.

References and Notes

1. K. Maney, *The Maverick and His Machine*, John Wiley & Sons, 2013, pp. 3–6, 11, 241–242.
2. W.A. Friedman, "John H. Patterson and the Sales Strategy of the National Cash Register Company, 1884 to 1922," *Business History Rev.*, vol. 72, no. 4, 1998, pp. 522–584.
3. F.G. "Buck" Rodgers, *The IBM Way: Insights into the World's Most Successful Marketing Organization*, Harper & Row, 1986, pp. 73–80.
4. W.D. Jones, "Personal Observations and Comments Concerning the History and Development of IBM: The Perspective of Walter D. Jones," *The IBM Century: Creating the IT Revolution*, J.R. Yost, ed., IEEE CS Press/John Wiley & Sons, 2011, p. 42.
5. That growing body of sales information and research is described in W. Friedman, *Birth of a Salesman: The Transformation of Selling in America*, Harvard Univ. Press, 2004, pp. 151–171.
6. For an example of such a pep talk, see Friedman, *Birth of a Salesman*, pp. 79–80.
7. For an account of my time as a salesman, see J.W. Cortada, "'Carrying a Bag': Memoirs of an IBM Salesman, 1974-1981," *IEEE Annals of the History of Computing*, vol. 35, no. 4, 2013, pp. 32–47.
8. Conversation between J.W. Cortada and P. Cushing (and several members of her staff), 12 Mar. 2009; notes in author's personal files.



James W. Cortada is a senior research fellow at the Charles Babbage Institute at the University of Minnesota. He was employed at IBM from 1974 to 2012. Contact him at jcortada@umn.edu.

Andrew V. Haeff: Enigma of the Tube Era and Forgotten Computing Pioneer

Jack Copeland
University of Canterbury, New Zealand

Andre A. Haeff
A.V. Haeff Papers

Prolific, yet neglected inventor Andrew Vasily Haeff (1905–1990) made numerous contributions to vacuum tube art, including the traveling wave tube, the inductive-output tube, the electron-wave tube (or "double-stream" amplifier), the resistive wall tube, and many others. Haeff's contributions to computing history include his pioneering computer monitor technology, his high-speed electrostatic computer memory tube, and his early work in the display and storage of text and graphics.

Harbin to Caltech

Andrei—later Andrew or Andy—Haeff was born in Moscow on 12 January 1905 (or 30 December 1904, by the Julian Calendar in use in Russia at the time).[1] In 1920 his family fled from Russia to Harbin, in northeastern China. An important railway city not far from the Russian border, Harbin became home to a large Russian population in the wake of the 1917 Bolshevik revolution. Haeff's father, Vasili, owner of the Churin Trading Company and a venture capitalist specializing in gold mining, continued the family business from Chinese soil. Haeff completed high school at Harbin and went on to study electrical and mechanical engineering at the Russian Polytechnic Institute there, graduating in January 1928.[2] Later that year, following a successful application to the California Institute of Technology, he came to the United States. At Caltech, Haeff obtained a master's degree in electrical engineering in 1929 and a PhD in electrical engineering and physics in 1932. Caltech's director Robert A. Millikan introduced the young Haeff to Einstein as one of the institution's most promising graduate students.

The topic of Haeff's doctoral dissertation was "Ultra High Frequency Oscillators," in particular the 1,000 MHz oscillator that he had developed.[3] He was soon using this oscillator in a transmitter that he built in order to carry out UHF communications experiments involving his first major invention, the traveling wave tube (TWT) amplifier.[4] The TWT went on to become one of the most important paradigms in microwave engineering, with interest in the tube remaining strong today, especially for radar and communications applications.[5]

Haeff's younger brother, Alexei Haieff, followed him to the US in 1931. Haieff, a talented composer, soon gained a reputation in his adopted country, spending long periods with Igor Stravinsky in Hollywood. It was a favorite family anecdote that when Haieff and Stravinsky were driving back to Hollywood together through the Rocky Mountains, Stravinsky exclaimed irritably, "I despise mountains—they don't tell me anything."[6]

Andrew V. Haeff

Born: 12 January 1905, Moscow, Russia

Died: 16 November 1990, Whittier, California

Education: BS (electrical and mechanical engineering), Russian Polytechnic Institute (Harbin, China), 1928; MS (electrical engineering), California Institute of Technology, 1929; PhD (electrical engineering and physics), California Institute of Technology, 1932.

Professional Experience: RCA, vacuum tube research engineer, 1934–1941; Naval Research Laboratory, consulting physicist, 1941–1950; Hughes Electron Tube Laboratory, Hughes Aircraft Company, research director, 1950–1954; Hughes Research Laboratory, vice president and director of research, 1954–1961; Acoustica Associates, consultant, 1962; TRW, researcher, 1968–1975; Caltech and NRL, consultant, 1975.

Awards: IEEE Harry Diamond Memorial Award, 1950.

 Published by the IEEE Computer Society

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

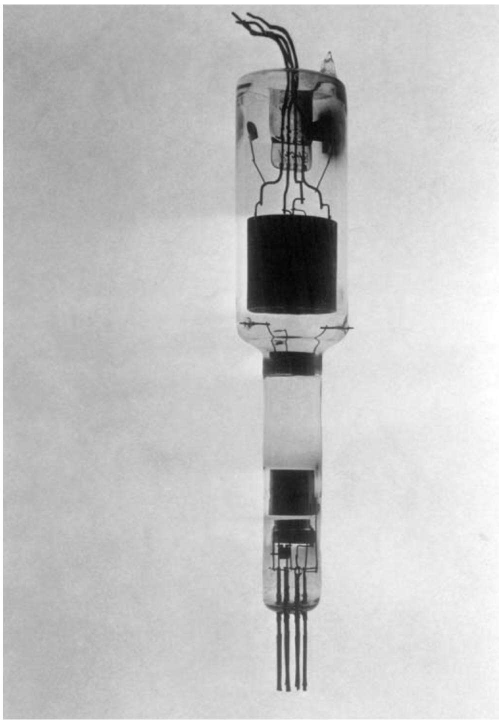


Figure 1. Prototype of Haeff's IOT (inductive output tube) in 1939. The market version of the tube was the RCA-825 Inductive-Output Amplifier.

Andrew too was highly musical, playing violin and piano from an early age and remaining musically active until almost the end of his life. At Harbin the young brothers would fill the house with music, Alexei at the piano and Andrei playing his violin inventively and with consummate style. In his teens Andrei was, like his brother, an avid composer, and dreamed of writing music to accompany Hollywood's silent movies. He even mailed a proposal from China to the Metro-Goldwyn-Mayer studios in Hollywood, but fortunately for the vacuum tube industry, nothing came of it.

Radio Corporation of America, 1934–1941

In March 1934, Haeff shifted from academia to industry, joining the Research and Engineering Department of RCA in Harrison, New Jersey. He worked on the development of small tubes ("acorn" tubes) for use in television—the RCA passion—and on UHF transmitting and receiving tubes and circuits.[7] At RCA Haeff filed a number of patents on velocity-modulated tubes and embarked on an analysis of space-charge effects in magnetically focused electron beams, an investigation that laid the foundations for much of his life's work.[8]

He also secured the foundations of his personal life, marrying Sonya in 1936. Their only child, Andre, was born in 1938. It was at RCA that Haeff made his second major contribution to tube art. He realized in 1938 that a high-velocity bunched electron beam would generate electromagnetic energy if passed through a resonant cavity and, in 1939, filed the first IOT (inductive output tube) patent, describing the density-modulation induction phenomenon and the resultant amplification of very high frequencies.[9] In this patent he called his grid-controlled tube simply an "electron discharge device," using the phrase "inductive-output tube" in his 1940 IRE article on the tube.[10] (In the summer of 1940, Haeff took the train all the way from New Jersey to Los Angeles to present his invention at the IRE Pacific convention.[11]) Employing a system of short magnetic lenses, his IOT provided power amplification over a wide band of frequencies in the UHF range. The production version of Haeff's tube was the RCA-825 Inductive-Output Amplifier (see Figure 1).[12]

Haeff's IOT was used in RCA's historic 1939 demonstration of television's potential, a large-scale experiment involving scheduled TV broadcasts to metropolitan New York, using a transmitter on top of the Empire State Building.[13] Early live programs included coverage of President Roosevelt opening the 1939 New York World's Fair. Haeff's IOT was used in the crucial repeater stations that relayed the signal beyond line-of-sight, extending the range of the experimental transmissions as far as Riverhead on Long Island, a total distance of 70 miles.[14] His IOT was "the only tube in existence in 1939 which made the television radio-relay system possible at that time," Haeff said.[15]

Soon, however, velocity-modulation tubes in the klystron family eclipsed the IOT. Forty years later, Haeff's IOT was rediscovered by Donald Preist and Merrald Shrader of Varian Associates.[16] Preist and Shrader noted that "Haeff's tube produced over 35 watts CW output at 500 MHz, a remarkable performance at the time."[17] The 1981 prototype of their improved version of Haeff's tube had a power output 1,000 times greater than the original tube.[18] Varian marketed the Preist-Shrader IOT under the trade name "klystrode" because (as Preist and Shrader observed) "between the anode and the collector, the

Haeff tube is similar to a klystron, while between the cathode and the anode, it closely resembles the tetrode."[19] In the 1980s and 1990s, the IOT rapidly took over the UHF TV market, and today Haeff's tube excels in digital TV broadcasting.

By the end of the 1930s, Haeff was totally Americanized and had virtually lost his Russian accent (very occasionally a trace of it would reappear, in moments of high stress). Millikan described him as "Russian in origin but completely American in outlook, personal appearance and bearing."[20] He was devoted to his new country. A dark, serious, thickset teddy bear of a man, Haeff was kind and friendly, but also shy. His favorite conversations were one-on-one, deep, and usually scientific. Haeff enjoyed the company of creative people, whether scientists, musicians, artists or writers. He was less fond of hidebound or dictatorial thinkers—especially the bureaucrats and administrators with whom he was increasingly forced to mix as he moved up the scientific career ladder. Ironically, he himself eventually became a top-level research administrator, a role he carried out with aplomb but did not much enjoy.

Naval Research Laboratory, 1941–1950

In 1941 the effort to develop radar was consuming ever-larger numbers of electronic engineers. Haeff entered the radar battle full time in March of that year, joining the staff of the Naval Research Laboratory (NRL) in Washington DC, with the rank of consulting physicist for the Radio Division.[15] "I felt that I could contribute considerably more if I worked directly for the Government on National Defense projects," he said.[15] Haeff played a significant role in the wartime development of radar and, in 1942, was a founder member of the legendary Vacuum Tube Development Committee (VTDC).

A glimpse of the nature and scope of Haeff's war work is provided by the flurry of patent applications that he lodged at war's end, during the period from September 1945 to February 1946. He was prolific in inventing new types of microwave signal generators and radar pulse generators, and he also contributed to radar countermeasures, inventing a sophisticated pulse-jamming system (with Franklin Harris). This rendered enemy radar equipment ineffective by transmitting interfering pulses.[21] The pulses were synchronized with, and powerful enough to obscure, the echo signals returning to the enemy receiver.

Haeff's pulse-jammer was designed for use against high-accuracy radar systems, such as shore-based fire control (gun-targeting) equipment. He spent many weeks during the early part of 1945 aboard a Navy cruiser off the California coast, testing the equipment in preparation for the planned (but pre-empted) invasion of Japan later that year. Of the five pulse-jammer patents that Haeff applied for in early 1946, three were withheld for security reasons until the 1970s, and his advanced design continues to be referenced in patents on jamming equipment up to the present day.

Haeff's UHF signal generators, developed at the NRL from 1943, generated radio frequency energy of known wavelength and amplitude.[22] They were used for testing and adjusting many varieties of radio equipment in the laboratory, factory, and field. The generators could deliver either continuous or pulsed output and, in pulse mode, were used principally to evaluate the performance of radar receivers and radar jammers. The Bureau of Ships and the Bureau of Aeronautics granted contracts to various manufacturers to produce signal generators according to Haeff's design (with Haeff acting as advisor to the manufacturers). Hundreds were produced for distribution to naval bases and to ships. One of the manufacturers was the fledgling Hewlett-Packard Company, situated in what is now Silicon Valley. Hewlett-Packard turned out Haeff's signal generators for the Navy and the Army, as well as for Britain and Russia. Commercial signal generators manufactured during the 1950s by RCA, General Communications, and Airadio as well as Hewlett-Packard (notably the Hewlett-Packard Model 610A), all followed Haeff's designs closely.[23] It was the Haeff signal generator, and also his radar jammer, that first put Hewlett-Packard on the industrial map.[24]

The war over, Haeff became head of the new Vacuum Tube Research Section at NRL, where in addition to directing tube research, he made his next two major contributions to the art. Pursuing his earlier study of space-charge effects, he investigated the interaction of electron streams of different velocities and discovered a new means of amplifying microwave energy.[25] Haeff called his new microwave amplifier the "electron-wave tube," now known as the double-stream amplifier. This was effectively a traveling wave tube with the helical electrode replaced by a second electron beam. Haeff conceived the basic idea in April 1946,[26] and in December of that year, he gave the first written description of the tube in a brief two-page note.[27] The concept of the double-stream amplifier had arrived, and to the proud Haeff, it looked a wonderful idea. He eventually reduced the idea to practice in May 1948.[26]

Phenomena discovered in tube research often proved to be of importance in astrophysics, and Haeff's double-stream instability was no exception. Haeff himself, in a foray into astrophysics, suggested that his double-stream effect accounted for the origin of solar radio noise, conjecturing that intermingling streams of charged particles emitted by the sun will greatly amplify an initial disturbance.[28] He also suggested that the Aurora Borealis (Northern Lights) and Aurora Australis are produced by a release of energy from streams of solar electrons that are pulled in by the Earth's magnetic field. Haeff's double-stream instability remains the subject of fundamental research today, in connection with particle accelerators and high-energy electronics, for example.

It was also during his highly productive period at the NRL that Haeff invented his "Memory Tube," the basis of his various contributions to computing history.[29] The Memory Tube stored information on a coated glass screen. Haeff had a prototype of this electrostatic tube working in June 1947[30] and, in the same month, prepared a confidential report on the tube for NRL. In this he wrote, "The signal-storage device described in this paper has many applications. It promises satisfactory solutions to such problems as … flash communications, the storage and reading of binary numbers for electronic digital computers, and many other problems."[30]

An important piece of early computer technology, the Memory Tube had its origins in Haeff's radar research. The tube would, he said, permit "simultaneous multicolour and three-dimensional presentation of radar or sonar data," moreover offering operators the advantage of "daylight viewing," and the tube could automatically generate a trace whose length was "proportional to the velocity of the target."[31] The tube was quickly declassified by NRL, and Haeff presented it at the IRE Electron Tube Conference in Syracuse, New York, on 10 June 1947.[32]

His invention aroused considerable interest. The Memory Tube was written up in *Newsweek* in September 1947 and then in *Popular Science* in May 1948.[33] "The Navy's new electronic memory tube," *Popular Science* reported, "remembers signals as long as you want it to."[34] *Newsweek* described the Memory Tube as a "long-sought memory device for the new 'electronic brain' calculating machines now being designed as successors to the Eniac," and the article echoes Haeff's view that the tube could resolve what *Newsweek* called the memory "bottleneck"—the problem of developing a fast, cheap memory capable of keeping pace with the high-speed electronic processors then under consideration.[35]

With one of the world's first electronic storage and display devices functioning in his laboratory, it is unsurprising that Haeff also became a pioneer of electronic graphics. He was probably the first to store graphics and text for a prolonged period on an electronic visual display screen, using the Memory Tube to display letters and pictures early in 1947. He also took the first historic steps toward digital graphics and text, storing letters and images on the tube's electrostatic screen in the form of discrete luminous elements (picture elements, or "pixels").

As a high-speed computer memory, Haeff's Memory Tube was eclipsed by the British Williams tube, although Haeff-type tubes did form the main high-speed memory of MIT's Whirlwind I computer. The Memory Tube had its most significant impact on computing as a display device. Hughes Products commercialized the Memory Tube, marketing versions of it called the Memotron, used for storing and displaying graphics, and the Typotron, which functioned as a text-based computer output device. Later forms of Haeff's Memory Tube were in common use as computer monitors and interactive graphics terminals until the 1980s, most notably the big screen Tektronix 4014, which many will remember as a thoroughly modern alternative to interacting with a mainframe via a paper-fed teletype.

Haeff was the first recipient of the IEEE Harry Diamond Memorial Award, presented to government servants for "outstanding technical contributions" (www.ieee.org/about/awards). The citation read, "For his contribution to the study of the interaction of electrons and radiation, and for his contribution to the storage tube art."[36] The award, bestowed in 1950, was primarily for Haeff's ground-breaking inventions, the Memory Tube and the double-stream amplifier,[37] but undoubtedly his radar signal integrating tubes, signal generators, and radar jammers also played a significant role in determining his selection for the award.

Hughes Aircraft Company, 1950–1961

The pivotal year 1950 saw Haeff leaving the NRL for the Research and Development Laboratories of the rapidly expanding Hughes Aircraft Company in Culver City, California. Owned by Howard Hughes, the flamboyant and eccentric entrepreneur, aviator, and movie producer, the company was at that time primarily under the technical leadership of Simon Ramo and Dean Wooldridge. Electronics was the main focus at Hughes and the company was hiring stellar researchers. The Hughes strategy was to take on military-oriented research problems that were sufficiently hard to deter competitors.[38] When his friend Si Ramo first broached a move to Hughes in January 1950, Haeff was initially reluctant to leave the NRL, but things changed following the outbreak of the Korean War later that year. Hughes was a leading supplier to the US forces and, by 1957, was the largest defense contractor in the United States. Haeff's new position, from November 1950, was head of the Hughes Electron Tube Laboratory, a move that cemented his transition from researcher to research director.

Haeff set up the Electron Tube Laboratory and led the Hughes research program in storage tubes. Under his direction, the Electron Tube Laboratory developed his Memory Tube and his TWT. Haeff continued to invent, especially in the field of microwave amplification, devising first his electron-stream amplifier tube (filing a patent in April 1952), and later the resistive-inductive wall amplifier, or "resistive-wall amplifier," with his Electron Tube Laboratory colleague Charles Birdsall (they filed for the first patent in October 1952).[39] The resistive-wall amplifier exploits an instability occurring when an electron beam flows close to a coated surface. The resistive-wall instability discovered by Haeff and Birdsall is now the subject of a considerable literature, especially in connection with plasma work and high-energy particle accelerators.

In 1954, Haeff's Electron Tube Laboratory was merged with other Hughes laboratories to form a single entity under Haeff's overall control, the Hughes Research Laboratories.[40] At this time Haeff was made a vice president of Hughes and designated director of research (see Figure 2). In a large, research-heavy organization such as Hughes—the company had a workforce of over 20,000 by 1955[41]—this was a superb, if demanding job.

Figure 2. Haeff in brother Alexei's Manhattan apartment in 1954.

Haeff's traveling wave interaction was core to the early maser developments ("microwave amplification by stimulated emission of radiation"). Later, his double-stream amplifier was important in the double beam cyclotron maser, essentially a double-stream amplifier in which the two electron beams travel at relativistic speeds.[42] Following the pioneering invention of the ammonia beam maser by Charles Townes at Columbia University in 1954, Haeff had a hand in setting up the maser research program at Hughes. The laser (or optical maser) was developed in Haeff's Research Laboratories by Ted Maiman in 1959, and the prototype Maiman ruby laser was built and operated in Haeff's laboratory during 1960. Haeff took color slides home to show his excited family. He himself went on to invent a number of laser devices, including an influential chemical laser apparatus in 1968. This achieved radiation amplification by siting a chemically reacting mixture next to an optical cavity.[43]

The laser opened up new horizons in military electronics. An internal memo to Haeff, written a few weeks after the ruby laser first operated, suggested researching its potential use in "communication systems, optical radars, passive detectors, … destructive weapons, … submarine detection, inertial guidance."[44] Haeff gladly harnessed his

Haeff is among America's most brilliant inventors, yet his name is little known even within the electronic engineering community.

brilliant mind for the production of military hardware. A technologically superior force had overwhelmed the land of his birth in 1941, and from that year, he had devoted himself to the development of military and military-related technology for the defense of his adopted country.

The launch of the Russian Sputnik satellite in October 1957 triggered the lucrative space race, and in 1959, Hughes entered the field of space communications. As director of research, Haeff played a leading role in managing a research and development program that led rapidly to Hughes' Syncom, the first geosynchronous communications satellite. At Syncom's heart was a lightweight TWT developed in Haeff's laboratories. The research program midwifed by Haeff led to the first operational commercial communications satellite, the Syncom-based Early Bird (also known as Intelsat I), launched in 1965, and ultimately to Hughes' domination of communications satellite manufacture.[45]

The Later Inventions

Haeff himself left Hughes in 1961, following a period of illness and exhaustion. He bravely made the step back from director of research to researcher and inventor, working on his own and also as an independent consultant. While consulting for Acoustica Associates (a firm led by his friend and former Hughes Vice President Andrew "Rus" Russell), Haeff invented his volumetric measuring device in October 1962, the time of the Cuban missile crisis. This important and influential device used sound waves to measure the volume of fuel in missiles, rockets, and spacecraft and was able to function in a zero gravity environment.[46]

The patents began to flow faster again, a diverse cascade of inventions. Still working with the properties of sound, Haeff carried out a series of experiments in 1963 (with Cameron Knox) showing that, under certain conditions, the human ear is able to perceive high frequency ultrasound as ordinary sound.[47] In 1964, now working from his home in West Los Angeles, Haeff became fascinated with the idea of using laser beams to scan 3D objects—statues, buildings, works of art, museum treasures—and to recreate the objects virtually. He foresaw industrial, household, and military applications for his new invention, a harbinger of virtual reality that he prosaically described in his 1964 patent application as an "Apparatus for Scanning and Reproducing a Three-Dimensional Representation of an Object."[48]

In 1959, while still research director and vice president at Hughes, Haeff began to study controlled nuclear fusion. He realized that since classical nuclear fission is environmentally hazardous, somehow the clean-burning hydrogen fuel of the sun would have to be harnessed, first in the laboratory and then in a revolutionary new type of power plant. He struggled intermittently with the problem of implementing fusion, envisioning the use of a plasma containment vessel. It was not until 1968 that he made a concrete contribution, when in a sustained burst of inspiration, he invented his plasma containment device just weeks after inventing the chemical laser apparatus described earlier.[49] Haeff was by this time employed in the research laboratory of the Thompson-Ramo-Wooldridge Corporation (TRW), whose founders were the same Simon Ramo and Dean Wooldridge who, 18 years earlier, had brought Haeff to Hughes from the NRL.

Haeff stayed at TRW until 1975, when he finally retired from formal employment. He continued consulting at Caltech and NRL, and (like Einstein before him) he became increasingly engrossed with a unified theory of gravity, hoping to integrate gravitation and quantum mechanics. Haeff labored on his unified theory at home, publishing some abstracts, but the work was left unfinished. He died in Whittier, California, on 16 November 1990.

Haeff is among America's most brilliant inventors, yet his name is little known even within the electronic engineering community.

Acknowledgments

Copeland is grateful to the following institutions for supporting this research: University

of Canterbury, New Zealand; University of Queensland, Australia; Federal Institute of Technology (ETH), Zurich, Switzerland; and Det Informationsvidenskabelige Akademi, Copenhagen University, Denmark.

References and Notes

1. This biography is the first published account of the enigmatic Haeff's life and remarkable work. Published source material on Haeff is scarce. There are four brief notes in the *Proc. IRE*, presumably drafted by Haeff himself: "Contributors to Proceedings of the I.R.E.: Andrew V. Haeff," vol. 27, 1939, p. 618; vol. 34, 1946, p. 864; vol. 37, 1949, p. 59; vol. 41, 1953, p. 938.
2. A.V. Haeff, "Career Summary, 1928–1950," p. 1 (in *A.V. Haeff Papers*).
3. A.V. Haeff, "*Ultra High Frequency Oscillators*," PhD thesis, California Inst. of Technology, 1932.
4. Haeff, "Career Summary, 1928–1950," p. 2. A.V. Haeff, "Device for and Method of Controlling High Frequency Currents," US patent 2,064,469, filed Oct. 1933, issued Dec. 1936.
5. Haeff's 1933 TWT amplifier is the topic of our forthcoming 2015 article in *IEEE Spectrum.*
6. A short biography of Alexei Haieff (by Andre A. Haeff) is contained in the sleeve notes for "Tiger of Harbin: Three Musical Works by Alexei Haieff," Alexei Records, 2010.
7. Haeff, "Career Summary, 1928–1950," p. 2. See, for example, P.D. Zottu, L.S. Nergaard, and A.V. Haeff, "High Frequency Power Tube," US patent 2,113,671, filed Dec. 1936, issued Apr. 1938.
8. A.V. Haeff, "Space-Charge Effects in Electron Beams," *Proc. IRE*, vol. 27, 1939, pp. 586–602.
9. A.V. Haeff, "Electron Discharge Device," US patent 2,237,878, filed Feb. 1939, issued Apr. 1941; A.V. Haeff, "An Ultra-High-Frequency Power Amplifier of Novel Design," *Electronics*, Feb. 1939, pp. 30–32; Haeff, "Career Summary, 1928–1950," p. 3.
10. A.V. Haeff and L.S. Nergaard, "A Wide-Band Inductive-Output Amplifier," *Proc. IRE*, vol. 28, 1940, p. 130. Nergaard, Haeff's assistant, carried out measurement work on the IOT under Haeff's supervision (Haeff, "Career Summary, 1928–1950," p. 3).
11. A.V. Haeff, "Ultra-High-Frequency Receiving and Transmitting Tubes," Inst. of Radio Engineers Pacific Coast Convention, Aug. 1940; Summary of "Ultra-High-Frequency Receiving and Transmitting Tubes," mailed to S.S. Mackeown at the Calif. Inst. of Technology, 10 June 1940 (in *A.V. Haeff Papers*).
12. K. Kilbon, "Television," *Pioneering in Electronics: A Short History of the Origins and Growth of RCA Laboratories, Radio Corporation of America, 1919 to 1964,* 1964, chap. 3; www.davidsarnoff.org/kil-chapter03.html. "RCA 825 Inductive-Output Amplifier," brochure, RCA Manufacturing Company, 1940 (in *A.V. Haeff Papers*); the brochure cites Haeff's articles "An Ultra-High-Frequency Power Amplifier of Novel Design" and "A Wide-Band Inductive-Output Amplifier," p. 8.
13. Kilbon, *Pioneering in Electronics*, chap. 3; Haeff, "Career Summary, 1928–1950," p. 3.
14. Haeff, "Career Summary, 1928–1950," p. 3; F.H. Kroger, B. Trevor, and J.E. Smith, "A 500-Megacycle Radio-Relay Distribution System for Television," *RCA Rev.*, vol. 5, 1940, pp. 31–50.
15. Haeff, "Career Summary, 1928–1950," p. 3.
16. D.H. Preist and M.B. Shrader, "The Klystrode—An Unusual Transmitting Tube with Potential for UHF-TV," *Proc. IEEE*, vol. 70, 1982, pp. 1318–1325. See also R.D. Peters et al., "The Klystrode IOT: A New, Compact, High-efficiency Electron Tube," *Microwave Processing of Materials IV*, M.F. Iskander, R.J. Lauf, and W.H. Sutton, eds., Materials Research Soc., 1994, pp. 97–108.
17. M.B. Shrader and D.H. Preist, "RF Sources for Particle Accelerators: A Progress Report," Varian, EIMAC Division, Oct. 1981, p. 2 (in *A.V. Haeff Papers*).
18. Shrader and Preist, "RF Sources for Particle Accelerators: A Progress Report," p. 3.
19. Preist and Shrader, "The Klystrode—An Unusual Transmitting Tube with Potential for UHF-TV," p. 1320.
20. R.A. Millikan to A.V. Haeff, letter, 9 Sept. 1933 (in *A.V. Haeff Papers*).
21. A.V. Haeff and F.H. Harris, "Pulse Transmission System," US patent 2,561,363, filed Jan. 1946, issued July 1951; A.V. Haeff and F.H. Harris, "Modulator," US patent 2,562,907, filed Feb. 1946, issued Aug. 1951; A.V. Haeff, "Pulse Generation System," US patent 3,624,652, filed Jan. 1946, issued Nov. 1971; A.V. Haeff and F.H. Harris, "Synchronized Pulse Jammer and Decoy," US patent 3,760,417, filed Jan. 1946, issued Sept. 1973; A.V. Haeff and F.H. Harris, "Automatic Power Control Pulse Modulator," US patent 3,579,238, filed Feb. 1971, issued May 1971.
22. Haeff, "Career Summary, 1928–1950," p. 4; A.V. Haeff, T.E. Hanley, and C.B. Smith, "Wide-Range Ultra-High-Frequency Signal Generators," *Proc. IRE—Waves and Electrons Section*, vol. 35, 1947, pp. 1137–1143; A.V. Haeff, T.E. Hanley, and C.B. Smith, "Ultra High Frequency Signal Generator," US patent 2,535,062, filed Apr. 1945, issued Dec. 1950; A.V. Haeff and C.B. Smith, "Ultra High Frequency Signal

Generator," US patent 2,727,181, filed Oct. 1945, issued Dec. 1955; A.V. Haeff and T.E. Hanley, "Microwave Signal Generator," US patent 2,696,554, filed Oct. 1945, issued Dec. 1954; A.V. Haeff and T.E. Hanley, "Signal Generator," US patent 2,589,248, filed Jan. 1946, issued Mar. 1952.

23. Haeff, "Career Summary, 1928–1950," p. 5.
24. D. Packard, *The HP Way: How Bill Hewlett and I Built Our Company*, Collins, 1995, p. 59.
25. A.V. Haeff, "Space-Charge Wave Amplification Effects," *Physical Rev.*, vol. 74, 1948, pp. 1532–1533; A.V. Haeff, "The Electron-Wave Tube—A Novel Method of Generation and Amplification of Microwave Energy," *Proc. IRE*, vol. 37, 1949, pp. 4–10.
26. A.V. Haeff to T. Hayward Brown (US Dept. of Justice), letter, 16 Feb. 1954 (in *A.V. Haeff Papers*).
27. A.V. Haeff, "Microwave Amplifier Tube," 21 Dec. 1946 (in *A.V. Haeff Papers*).
28. A.V. Haeff, "Space-Charge Wave Amplification Effects," *Physical Rev.*, vol. 74, 1948, pp. 1532–1533; A.V. Haeff, "On the Origin of Solar Radio Noise," *Physical Rev.*, vol. 75, 1949, pp. 1546–1551.
29. Haeff's Memory Tube is the topic of our forthcoming article "Screen History: The Haeff Memory and Graphics Tube," *IEEE Annals for the History of Computing.*
30. A.V. Haeff, "The Memory Tube," NRL Progress Report, Naval Research Lab., 10 June 1947, p. 1 (in *A.V. Haeff Papers*).
31. A.V. Haeff, "A Memory Tube," *Electronics*, Sept. 1947, pp. 80, 83; A.V. Haeff, "Method of Storing, Maintaining, and Reproducing Electrical Signals, and Means Therefor," US patent 2,813,998, filed Aug. 1947, issued Nov. 1957. The following two radar equipment patents are precursors to the Memory Tube patent 2,813,998: A.V. Haeff, "'Signal Integrating Tube for Radar," US patent 2,617,07, filed Jan. 1947, issued Nov. 1952; and A.V. Haeff, "Signal Integrating Tube," US patent 2,681,425, filed Jan. 1947, issued June 1954.
32. See A.V. Haeff, "The Memory Tube and its Application to Electronic Computation," *Mathematical Tables and Other Aids to Computation*, vol. 3, 1948, pp. 281, 286.
33. "Memory in a Tube," *Newsweek*, 15 Sept. 1947, pp. 53–54; "This Tube Remembers Whatever It's Told," *Popular Science*, vol. 152, May 1948, p. 167.
34. "This Tube Remembers Whatever It's Told," p. 167.
35. "Memory in a Tube," p. 54.
36. The award certificate is in the *A.V. Haeff Papers.*
37. Haeff, "Career Summary, 1928–1950," p. 7.
38. J.L. Bromberg, *The Laser in America, 1950–1970*, MIT Press, 1991, p. 25.
39. A.V. Haeff, "Electron Stream Amplifier Tube," US patent 2,740,917, filed Apr. 1952, issued Apr. 1956; A.V. Haeff and C.K. Birdsall, "Resistive-Inductive Wall Amplifier Tube," US patent 2,793,315, filed Oct. 1952, issued May 1957; C.K. Birdsall, G.R. Brewer, and A.V. Haeff, "The Resistive-Wall Amplifier," *Proc. IRE*, vol. 41, 1953, pp. 865–875.
40. G.F. Smith, interview by J.L. Bromberg, 5 Feb. 1985, Niels Bohr Library and Archives, Am. Inst. of Physics; http://aip.org/history/ohilist/4895.html.
41. Bromberg, *The Laser in America*, p. 26.
42. See, for example, G. Bekefi, "Double Beam Cyclotron Maser," US patent 5,363,054, filed Feb. 1992, issued Nov. 1994.
43. A.V. Haeff, "Chemical Laser Apparatus," US patent 3,575,669, filed June 1968, issued Apr. 1971.
44. Smith to Haeff, memo, 21 July 1960; Smith, interview by Bromberg, 5 Feb. 1985.
45. D.J. Whalen, "Billion Dollar Technology: A Short Historical Overview of the Origins of Communications Satellite Technology, 1945–1965," *Beyond the Ionosphere: Fifty Years of Satellite Communication*, A.J. Butrica, ed., NASA History Office, 1997.
46. A.V. Haeff, "Volumetric Measurement System," US patent 3,237,451, filed Oct. 1962, issued Mar. 1966.
47. A.V. Haeff and C. Knox, "Perception of Ultrasound," *Science*, vol. 139, 1963, pp. 590–592.
48. A.V. Haeff, "Apparatus for Scanning and Reproducing a Three-Dimensional Representation of an Object," US patent 3,636,250, filed Feb. 1964, issued Jan. 1972.
49. A.V. Haeff, "Plasma Containment Device," US patent 3,643,123, filed Oct. 1968, issued Feb. 1972.

Jack Copeland *is a distinguished professor at the University of Canterbury, Christchurch, New Zealand. Contact him at jack.copeland@canterbury.ac.nz.*

Andre A. Haeff *is an archivist for the* A.V. Haeff Papers*, Los Angeles. Contact him at alexeirec@aol.com.*

TENEX and TOPS-20

Dan Murphy

Editor: David Walden

In the late 1960s, a small group of developers at Bolt, Beranek, and Newman (BBN) in Cambridge, Massachusetts, began work on a new computer operating system, including a kernel, system call API, and user command interface (shell). While such an undertaking, particularly with a small group, became rare in subsequent decades, it was not uncommon in the 1960s. During development, this OS was given the name TENEX. A few years later, TENEX was adopted by Digital Equipment Corporation (DEC) for its new line of large machines to be known as the DECSYSTEM-20, and the operating system was renamed to TOPS-20.

I followed TENEX (or vice versa) on this journey, and these are some reflections and observations from that journey. I will touch on some of the technical aspects that made TENEX notable in its day and an influence on operating systems that followed as well as on some of the people and other facets involved in the various steps along the way.

The relationship between BBN and DEC is a significant part of the story. In 1961, BBN was one of the first sites external to DEC to install and use a PDP-1 (DEC's first computer), and it did some early experiments in timesharing using this machine and some locally designed modifications. During the subsequent years, many discussions took place between the two companies about the best way to design and build computers.

I was an undergraduate at MIT from 1961 to 1965. After receiving my degree in 1965, I went to work for BBN doing various systems programming activities and, in particular, supporting the LISP system then in use for AI research. We ran LISP on the PDP-1 and, because of the needs of the users, wound up doing considerable development on the LISP system itself. These included (1) converting our single-user PDP-1 LISP system into a LISP timesharing system and (2) adding paging capabilities to the LISP system to give the effect of much more memory than the PDP-1 actually had.

From the time I arrived at BBN, our group was eager to move up to a PDP-6, DEC's much larger machine, one that seemed especially suited for supporting large LISP projects. From our own work, however, we also felt that virtual memory and effective timesharing were essential, and we had a number of discussions with DEC engineers and management about hardware enhancements to the PDP-6 that would support these features.

As noted, we had developed both virtual memory and timesharing on our PDP-1 LISP system. This system used a high-speed drum as backing store for main memory and divided both main memory and the drum into fixed size units (pages) that could be copied back and forth as necessary. The LISP system itself used 18-bit pointers and was built as if there were an 18-bit (262,000 word) memory on the machine. However, with software paging, each reference to the target of one of these pointers was changed to a subroutine call (or in-line sequence). The subroutine would take the high order few bits of the pointer as an index into an array (the page table), which would contain the current location of the page. If the page were then in main memory, the page number bits from the table would replace those that had been used as the index into the table, and the reference would be completed. If the desired page were not in main memory, a call would be made to a "page manager" routine to rearrange things as necessary. That meant selecting a page in main memory to be moved (copied) back to the drum and then reading into this page the desired page from the drum. Finally, the table would be adjusted to reflect these changes, and the reference sequence begun again.

In actual use, this system was fairly effective. The reference patterns of the typical LISP programs that we were running were such that the system was not dominated by waits for drum I/O. That is, most pointer references were to pages then in main memory, and only a small fraction were to pages that had to be read in from the drum. Several studies, some by us at BBN, were done to investigate memory reference behavior and gather statistics from various programs.[1,2]

However, the actual number of references was sufficiently high that a great deal of time was spent in the software address translation sequence, and we realized that, ultimately, this translation must be done in hardware if a truly effective paged virtual memory system were to be built.

Deliveries of the PDP-6 began in 1964, and it immediately became the machine of choice for LISP programming. A single 36-bit word holding two 18-bit addresses (a CAR and CDR in LISP terms) seemed designed especially for LISP (this wasn't mere coincidence). The big (for the time) address space of 2^{18} words offered the promise of fast, efficient execution of large LISP programs. Although the basic instruction set didn't include a CONS instruction (the LISP primitive that builds lists), one early PDP-6 had a special modification installed to provide that operation. This was the machine that was installed at the AI Lab of Stanford University, headed by one of the inventors of LISP, John McCarthy.

DEC Cancels the 36-Bit Architecture, or Does It?

The group at BBN had every intention of acquiring a PDP-6 to improve our LISP capabilities, but of course, we wanted it to have hardware paging so that we could bring over the paging techniques from our PDP-1 LISP. Several discussions were held between BBN and DEC, including Win Hindle and Alan Kotok on the DEC side. BBN was lobbying for paging to be included in a subsequent PDP-6 model, or possibly as an add-on, but these discussions came to an end when DEC announced in 1966 that there would be no further PDP-6 models—that is, that DEC was going out of the 36-bit business. This turned out to be the first of several such occasions.

Taking DEC at its word, BBN turned its attention to selecting another machine with a more promising future that we could use to support our various research projects. The result of this selection process was the purchase of an SDS-940. Scientific Data Systems (SDS) in El Segundo, California, was later acquired by Xerox and known as XDS. The SDS-940 was a 24-bit word machine with a modest virtual address space, but it did have hardware paging capabilities. Also, SDS was touting as a follow-on, the Sigma-7 then under development, which would be larger, faster, and do all the things we could ever possibly want. If it did, we never found out because, by the time it came out, DEC had resurrected the 36-bit line, and we happily switched back to the architecture we really wanted.

It was also a factor that the Sigma-7 suffered a bit of schedule slippage. Yes, this actually happened despite firm assurances from SDS that no such thing was possible. SDS had even published several ads in trade magazines touting their forthcoming system as "the best timesharing system not yet available." Another memorable ad read, "They said we must be crazy to publish a firm software development schedule, but here it is." Apparently, "they" were right because that firm schedule went the way of most "firm" software development schedules of the period.

The former ad also inspired someone at DEC to create a counter ad for DEC's PDP-10 operating system, TOPS-10, which read "announcing the best timesharing system very much available." This was after the reborn 36-bit machine had been announced and named the PDP-10, aka DECsystem-10.[3]

BBN did use the SDS-940 for a couple of years however, and we ran on it an operating system developed by a group at University of California, Berkeley.[4] That was significant because a number of features in the Berkeley timesharing system were later modified and adopted into TENEX.

DEC's reentry into the large machine business was heralded by the announcement of the PDP-10 in 1967. Our group at BBN had learned of the project somewhat earlier, and we once again lobbied for the inclusion of a hardware paging system. And once again, this was to no avail. One advancement in the KA10, the first CPU model of the PDP-10 line, was the dual protection and relocation registers (the PDP-6 had only a single pair). This allowed programs to be divided into two segments: a reentrant, read-only portion and a read-write data area. That, however, was as far as DEC was willing to go at that time in advancing the state of the art of operating system memory management support.

Another factor was DEC's firm intent to keep down the price of this new large machine. DEC was doing well with small machines (PDP-5 and PDP-8 in particular), but the PDP-6 had suffered numerous engineering and reliability problems, it's price made it hard to sell, and some say, it almost wrecked the company. The KA10 was designed around a different hardware technology that ultimately proved reliable and easy to service in the field. The KA10 was also designed to be more affordable, including a requirement that a minimal machine could be configured and sold for under $100,000. This configuration was limited to 16,000 36-bit words of main memory and had no solid-state registers for the 16 accumulators (all AC references went to main memory at considerable cost in time). To support this, a "tiny" build of TOPS-10 was designed that would fit in 8,000 words, and various utilities were similarly squeezed: 1 K-word PIP and so on. Whether the $99,999 entry price was a good marketing ploy is hard to say. In any case, none of that configuration was ever sold.

Then too, paging and "virtual memory" were still rather new concepts at that time. Significant commercial systems had yet to adopt these techniques, and the idea of pretending to have more memory that you really had was viewed skeptically in many quarters within DEC.

Paging Requirement Leads to New Operating System

Undaunted by DEC's refusal to see the wisdom of our approach, BBN nonetheless planned the purchase of several KA10's and set about figuring out how to turn them into the system we really wanted. At the core of that system was efficient demand paging and sharing through use of memory mapping. As noted earlier, the KA10 had two "protection and relocation registers." These supported a virtual memory of sorts for user programs, although not a paged one. Hence, the operating system was required to shuffle large blocks of memory around as user programs grew and shrank in order to form contiguous areas. Also, there was no ability to share memory between user processes beyond the one segment that was reserved as a read-only area for program text.

In addition to the 36-bit architecture we favored, the PDP-10 system had good hardware modularity, including in particular the memory bus with independent memory units. This led us to conceive of the idea of putting a mapping device between the processor and the memories that would perform the address translation as we wanted it done. The mapping device became known as the "BBN Pager", and its registers, pointer types, and so on are described in earlier work.[5]

We also had to decide whether or not to attempt to modify TOPS-10 to support demand paging. The alternative was to build a new system from scratch, an ambitious undertaking even in those days. Obviously, we decided to build a new system—the system that was later named TENEX. This decision was justified largely on the grounds that major surgery would be required to adopt TOPS-10 as we desired, and even then we probably wouldn't be able to solve all the problems. In retrospect, this view was probably justified because, although TOPS-10 development continued for nearly 20 years after we started TENEX, TOPS-10 was never modified to have all the virtual memory features of TENEX/TOPS-20. TOPS-10 did ultimately support paging and virtual memory, but not the various other features.

Beyond that, there were a number of other features not related to paging that we wanted in an operating system. This further tilted the decision toward implementation of a new system. On the other hand, we had no desire or ability to implement new versions of the many language compilers and utilities then available under TOPS-10, so a method of running these images "as is" was needed. We decided it would be possible to emulate the operating system calls of TOPS-10 with a separate module that would translate the requests into equivalent native services. The plan then would be to implement a new system from the operating system interface level down, a user-friendly command language (shell), and particular major systems such as LISP that were key to our research work.

Like most other software systems, TENEX was an amalgam of ideas from a number of sources. Most of the features that came to be prized in TOPS-20 had roots in other systems. Some were taken largely unchanged, some were modified in ways that proved to be critical, and others served merely as the germ of an idea hardly recognizable in the final system. Among those that we can trace to one or more previous systems are "escape recognition," virtual memory structure, process structure and its uses, and timesharing process scheduling techniques.

Three system most directly affected the design of TENEX: the MULTICS system at MIT, the DEC TOPS-10 system, and the Berkeley timesharing system for the SDS 940 computer. MULTICS was the largest and most state-of-the-art system of that time, and it incorporated the latest ideas in operating system structure. In fact, it was popular in some circles to say that, with the implementation of MULTICS, "the operating system problem had been solved." Several members of the MIT faculty and staff who had worked on MULTICS provided valuable review and comment on the emerging TENEX design.

Many of the paging concepts came from our own previous work, the PDP-1 LISP system in particular. Other ideas had recently appeared in the operating system literature, including the "working set" model of program behavior by Peter J. Denning.[6] The TENEX paging design, supported by specific features

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

in the BBN Pager, allowed the operating system to track the working set of each process and to use an LRU algorithm when selecting pages for removal from main memory.

MULTICS had developed the concept of segmentation and of file-process mapping—that is, using the virtual address space as a "window" on data that permanently resides in a file. The TENEX file system[5] had a somewhat simplified implementation of this concept, with "shared" and "indirect" pointer types allowing the physical address of a page to be maintained in one place regardless of how many times it was mapped into user address spaces.

The Berkeley 940 system had a multiprocess structure, with processes that were relatively inexpensive to create. That in turn allowed such things as a system command language interpreter (shell) that ran in unprivileged mode and a debugger that did not share address space with the program under test and therefore was not subject to being destroyed by a runaway program.

These two concepts, virtual memory and multiple processes, were fundamental to the design of TENEX, and ultimately were key to the power and flexibility of TOPS-20. The two abstractions also worked well together. The concept of a process included a virtual address space of 262,000 words (18-bit addresses)—the maximum then possible under the 36-bit instruction set design. Extended addressing was still many years away, and the possibility that more than 262,000 words (about 1.3 Mbytes) might ever be needed rarely occurred to anyone.

We merely wanted the full virtual address space to be available to every process, with no need for the program itself to be aware of demand paging or to invoke a "go virtual" mode. We believed that mechanisms could be built into the paging hardware and operating system to track and determine process working sets, and it would do as good a job at managing memory efficiently as if the program explicitly provided information (which might be wrong). It might happen, of course, that a particular program had too large a working set to run efficiently in the available physical memory, but the system would still do the best it could and the program would make some progress at the cost of considerable page thrashing. In any event, the implementation of virtual memory was to be transparent to user programs.

TENEX was "born" as reported in a BBN internal newsletter from mid-1970: "TENEX was put officially on the air on June 15, 1970 with capabilities to serve LISP and machine language programmers. TENEX was scheduled to be on the air May 1, 1970. This six week slippage was the only unrecognized slip in a very tight schedule held since November, 1969."

So although we can't claim to have had no schedule slips, this announcement does reflect that it was about six months from the time the first line of code was written to when the system was sufficiently operational to begin supporting general users. Even at the time, that did seem remarkable.

Of course, the system was far from finished or presenting all the features we planned, and the stability did leave something to be desired. As the newsletter further reported, "While the crash rate was very high in the first week of operation, system's personnel were able to find and correct many bugs which were provoked by having 'real users' on the system… Reliability has continued to improve, with only 3 crashes occurring during the period July 1 through July 9."

Active development of TENEX continued for a number of years, but the original goals were largely achieved by the time we presented a technical paper[7] on TENEX at the ACM Symposium on Operating Systems Principles in October 1971. At the 2013 Symposium, this paper was given the SIGOPS Hall of Fame Award (www.sigops.org/award-hof.html), a recognition of the influence that TENEX had on subsequent operating system development. That paper discusses in detail many of the technical features that I have alluded to here.

A much earlier indication of the effectiveness of TENEX was apparent in the first few years however. Between 1970 and 1972, TENEX was adopted and installed by a number of other labs supported by the Advanced Research Projects Agency (ARPA) of the US Department of Defense—not surprising, as the needs of these groups were similar to those of the BBN AI research programs.

During this same period, the ARPA network was being developed and came on line. TENEX was one of the first systems to be connected to the ARPANET and to have OS support for the network as a general system capability. This further increased its popularity.

TENEX Moves to DEC

The circumstances under which TENEX moved to DEC and became TOPS-20 seem in retrospect to have included a number of fortuitous events. As noted earlier, by 1972, TENEX had achieved a certain amount of popularity among researchers on the ARPA network. The reborn 36-bit machine, newly christened the DECsystem-10 was enjoying reasonable success in a number of other markets as well. The prestige of being the choice of leading-edge researchers was worth advertising though, so DEC ran an ad in a number of trade publications headlined "ARPA has a network of Supercomputers" and pointing out what a large fraction of those were DECsystem-10s. In fact, most of those were running TENEX. By April 1972, there were seven sites in addition to BBN running TENEX.

BBN had a modest business building the outboard paging hardware that, with the technology of that day, required an entire tall 19-inch wide cabinet of logic. DEC, meanwhile, had begun work on a successor machine to be known as the KI10 (the "I" at least suggesting the IC technology that was to be used). As early as June 1970, meetings were held where BBN people attempted to persuade DEC to include paging hardware similar to the design of the BBN pager. Eventually, DEC decided to include paging in the KI10, but it was based on a much simpler architecture. DEC engineers were not convinced that the several forms of pointers (private, shared, indirect) and the core status table would be worth the amount of hardware required to support them. Nonetheless, they did choose the same page size, 512 words, which at least left open the door to some sort of later accommodation.

KI-TENEX

When the KI10 came out, DEC was disappointed (to say the least) by the lack of interest among the research community that had helped spark KA10 sales. The problem was that the machine would not run TENEX. It was almost twice as fast as the KA10, but the paging was different from what TENEX required. As a further irony, the version of TOPS-10 initially shipped with the KI10 used the paging hardware only to simulate the protection and relocation hardware of the KA10 and realized no benefit from it.

During the summer of 1972, I had decided to look for new opportunities outside of BBN. Not surprisingly, one company I talked to was DEC. In the course of those discussions, I was asked about the possibility of putting TENEX on the KI10. This was not a desire that was widespread within DEC in general or within DEC software engineering in particular, but it was of great interest to Allan Titcomb, who was the marketing manager covering the scientific and research markets. Titcomb wanted very much to sell some KI10's to the sites that were running TENEX.

The outcome of this was that I went to work for DEC, as a contractor. I contracted with DEC for a fixed-time (three months), fixed-price contract to make TENEX run on the KI10.[8] Compared to the commitment implied by starting and staffing a real OS development project, a one-man three-month contract must have seemed an acceptably small risk.

At the beginning of October 1972, I took my leave from BBN and settled into an office on 3-5 (building 3, floor 5) in DEC's original buildings in Maynard, Massachusetts. Being at the time still rather youthful in outlook, the idea of a three-month, fixed-price, one-man contract to port a base operating system to a new processor in a family didn't strike me as particularly scary. It helped that I also had an offer from DEC of a permanent position as an engineer in the TOPS-10 group after the contract was concluded.

As part of my send-off from BBN, a couple of coworkers who had previously worked at DEC gave me some farewell presents that, they assured me, would prove useful at DEC: a flyswatter and a can of bug spray. DEC's facilities in Maynard at the time lacked some of the aseptic uniformity generally expected of hi-tech offices and labs. Because of their age, history, and proximity to a mill pond and stream, the buildings were well supplied with various insect life and spiders, and my friends at BBN wanted to be sure I knew what I was getting into. Ultimately, I spent only a little over a year working in the Maynard mill buildings, but there were numerous occasions late in the evening when the possibility of further concentration on code was nil, and I found myself watching at length as a particularly skillful spider spun a nearly perfect circular web among the posts of my partition.

Paging Algorithm in Software

My plan for KI-TENEX did not involve major alterations in the paging structure of TENEX in order to deal with the simpler pager of the KI10. Rather, the idea was to simulate most of the logic of the BBN pager in software and use a KI10 page table only as a software cache or translation buffer for current virtual-to-physical page mappings. Logically, this was much like the design used in many later processors where the logic would be realized in microcode and the storage in RAM.

Implementation of PDP-10 code to simulate the BBN pager was not a large or difficult task and took probably less than half the time of the project. In addition to paging, it was necessary to write drivers for the file and swapping devices then being shipped by DEC, neither of which had been used at BBN. Checkout of TENEX on the KI10 did, however, encounter one new and rather obscure logic error in the paging hardware that caused program errors and crashes at seemingly random moments.

Well before the end of the three month contract period, TENEX was running well enough on the KI10 to support building new versions of itself. During this period, the system was up and running for several hours each day on a machine in the basement of the mill, and a few curious employees came around to try it. One such person was Dave Braithwaite, then in the -10 benchmark group, who brought over various benchmarks and tests to try.

The contract formally ended (successfully) when I delivered an official set of tapes containing the TENEX sources and a bootable system at the contractually appointed time. This was somewhat academic at that point, however, because it was not by any means the end of TENEX-related activity at DEC.

During the time I was working on KI-TENEX, a new processor, the KL10, was under active development in hardware engineering. The "L" in KL10 was originally intended to mean "low cost" because the KI10 was perceived as being somewhat expensive. However, technology was providing opportunities to make a significant jump in performance and that ultimately was to be the salient feature of the KL10. The product-line managers were seeing opportunities to grow in the high end, so the stage was set to consider some major changes in capabilities.

IBM Makes Virtual Memory Legitimate

Quite possibly, the final fortuitous event involved in the DEC decision to take TENEX as the base for a DEC product happened not at DEC but at IBM. It was during this period, in the latter part of 1972, that IBM announced "virtual memory" systems for the 360/370 family.[9] Although this was not entirely unexpected, it provided a major shot of legitimacy for the concept of virtual memory in computer system products. I (and other TENEX proponents) had been actively promoting the virtual memory capabilities of TENEX, but it took the IBM announcement to prove that such capabilities could be a significant factor in large systems markets. This is rather ironic because the memory management architectures in TENEX/TOPS-20 and the IBM systems were quite different.

Soon, discussions were being held around the idea that the KL10 would be—not just a new CPU for the existing DECsystem-10 but the cornerstone of a new VM product family including both hardware and software architectures. Although I was part of only a few of those discussions, I still look back with amazement at the speed and confidence with which the decision was made to undertake such a major departure. The key individuals included Allan Titcomb, who had initiated the KI-TENEX project; Fred Wilhelm, the engineering manager of the KL10 project; Bill Kiesewetter, the marketing manager of the DECsystem-10 product line; and John Leng, the product line manager. We didn't convene task forces or study committees or waffle on the idea for a year or two. We met, considered the issues, and decided.

Thus, by the end of the three-month KI-TENEX contract, I had a new offer from DEC to join the KL10 group as project leader for a new operating system for the KL10 based on TENEX. By the time I started work at DEC as an employee on 2 January 1973, one additional engineer had been recruited to form the nucleus of a development group: Peter Hurley. The two of us set up offices on the fifth floor of Maynard mill building 5, in a group with the hardware engineers, the product-line marketing and management people, and vice president Win Hindle.

The new operating system group (the name TOPS-20 didn't come until it was almost time to ship the product) grew to four people during its first year. In addition to Peter Hurley and me, Arnold Miller and Len Bosack joined the group within the first few months. Tom Hastings, one of the original developers of TOPS-10, was also involved, serving briefly as the supervisor. Although it started as part of KL10 hardware engineering, this new group made the transition to its more logical place in software engineering after the first year.

Several others served as supervisor of the group between its formation and first ship. On several occasions, the group hired its new supervisor. To save time, an interview would be set up with several members of the group at once who would fire questions at the hapless candidate. If the candidate couldn't survive this, he clearly wouldn't last in the day-to-day functioning of the group. Managers weren't the only candidates who were given practical scrutiny. Judy Hall, the fifth member hired into the group, and others who joined during that period, were typically asked to bring samples of their code when they came for interviews.

How the VAX Almost Had 36 Bits

By 1975, it had become clear that the PDP-11 architecture had topped out. In particular, the address space was too limited to support many growing applications. An effort was started to build a new machine that would sell at the upper end of the PDP-11 price range and beyond and would be the growth path for -11 users. This new machine, code-named the UNICORN, was to be based on the 36-bit architecture because there was already a suite of software available for it. Several of the most senior engineers in the PDP-11 groups began coming to Marlboro, Massachusetts, to talk about building a small -10—small to our way of thinking, but large in comparison to the -11 line.

One upshot of this work was that design work we had done to expand the PDP-10 address space beyond 18 bits came under review. With more ambitious goals for business and performance, and a greater appreciation for the pain of running out of address space, the PDP-11 engineers insisted that the extended addressing design be enhanced to improve ultimate effectiveness. They convinced us that performance should not be compromised for reasons of compatibility or conversion as had been contemplated in the original design.

A new extended addressing design emerged that was a big improvement in the long run. It was, however, too late to be fully implemented in the initial KL10 product. The new design incorporated a 30-bit address rather than the 23 we had initially considered as a kind of stopgap.

Ironically, the UNICORN project never came to fruition. Within a relatively short time, a conclusion was reached that, even if a cheap 36-bit architecture machine could be built, it would not be "culturally compatible" with the PDP-11 software and applications and so would not meet the need. Instead, a core group was formed to design an architecture that would be culturally compatible with the PDP-11 but would eliminate the limitations of the PDP-11 address space. Eventually, the machine built as a result of this effort was named VAX-11, for Virtual Address eXtension of the 11. As we know however, the designers did a lot more than just extend the address space. What is not widely known is that, for a while at least, the VAX was planned to be a 36-bit architecture machine.

Conclusion

The early popularity of TENEX on the ARPANET was certainly a key to its later acceptance, transformation, and adoption as TOPS-20 by DEC. That popularity in turn seems to have been based not only on its technical features but also on the fact that it was developed within that community and was responsive to it. When TOPS-20 became a DEC product, it became part of a much larger market and thus less responsive to any particular segment of the market.

In addition to that, as interest in other computer architectures increased in the late 1970s, many research sites came to the conclusion that they did not want to be dependent on any one vendor for either hardware or software and, instead, wanted "open" systems that would be amenable to local modification and evolution. This of course led to a rapid increase in the use and popularity of UNIX. The fact that UNIX was implemented in a reasonably portable language (at least as compared with 36-bit MACRO) also encouraged its spread to new and less expensive machines.[10] If I could have done just one thing differently in the history of TENEX and TOPS-20, it would be to have coded it in a higher-level language. With that, it's probable that the system, or at least large parts of it, would have spread to other architectures and ultimately survived the demise of the 36-bit architecture.

In a way, it did survive the 36-bit hardware, although not as a practical system. Eventually, emulators were written for the PDP-10 instruction set, including the TENEX/TOPS-20 paging hardware. DEC eventually released the sources for TOPS-20 for experimental use, and it was brought up under emulation. As PC systems became progressively cheaper and more powerful, the day came (long ago) when a desktop PC system could run TOPS-20 faster than the fastest hardware system that DEC ever built.

References and Notes

1. D.G. Bobrow and D.L. Murphy, "Structure of a LISP System Using Two-Level Storage," *Comm. ACM*, vol. 15, no. 3, 1967, pp. 155–159.

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

2. D.G. Bobrow and D.L. Murphy, "A Note on the Efficiency of a LISP Computation in a Paged Machine," *Comm. ACM*, vol. 11, no. 8, 1968, p. 558.
3. The first CPU in this new line of systems was designated the KA10. Subsequent CPUs would include the KI-10, KL-10, and KS-10. The 36-bit architecture, first introduced on the PDP-6, was commonly called the PDP-10, following DEC's naming formula which began with the PDP-1, although DEC itself did not use that term in marketing materials or documentation. DEC's operating system for the DECsystem-10 was named TOPS-10 and was carried forward from its original implementation on the PDP-6.
4. B.W. Lampson, W.W. Lichtenberger, and M.W. Pirtle, "A User Machine in a Time-Sharing System," *Proc. IEEE*, vol. 54, no. 12, 1966, pp. 1766–1774; http://research.microsoft.com/en-us/um/people/blampson/02-UserMachine/WebPage.html.
5. D.L. Murphy, "Storage Organization and Management in TENEX," *AFIPS Proc. FJCC*, vol. 41, 1972; http://tenex.opost.com/fjcc72/.
6. P.J. Denning, "The Working Set Model for Program Behavior," *Comm. ACM*, vol. 11, no. 5, 1968, pp. 323–333.
7. D.G. Bobrow et al., "TENEX, A Paged Time Sharing System for the PDP-10," *Comm. ACM*, vol. 15, no. 3, 1972; http://tenex.opost.com/tenex72.txt.
8. D.L. Murphy, "Implementation of TENEX on the KI10," TENEX Panel Session, NCC, 1974.
9. VM/370 was made available in 1972 as part of IBM's "System/370 Advanced Function" announcement; see www.vm.ibm.com/history/.
10. K. Thompson, "Reflections on Trusting Trust," http://cm.bell-labs.com/who/ken/trust.html; originally published *Comm. ACM*, vol. 27, no. 8, 1984, pp. 761–763.

Dan Murphy *worked at BBN on the development of systems software supporting AI research from 1965 to 1973. In 1973, he joined DEC as the technical lead of the group developing TOPS-20, and his later activities at DEC included development projects on VAX-VMS. He subsequently held positions at the Open Software Foundation (OSF), EMC, and L3 Communications Klein Associates. Contact him at dan.murphy@dlmmx.com.*

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

Events and Sightings

Chigusa Kita, Editor
Kansai University

CBI–NSF Computer Security History Workshop

The Charles Babbage Institute, supported by the National Science Foundation's Trustworthy Computing program, is engaged in a multiyear research project, "Building an Infrastructure for Computer Security History." The project entails conducting 30 oral histories with computer security pioneers, creating knowledge networking resources, building archival collections, and preparing a set of scholarly publications. On 11–12 July 2014, CBI held a workshop to facilitate and advance scholarship and understanding of computer security history.

An open call for papers yielded high-quality proposals in a range of topics and themes—from computer crime, security metrics, standards, and encryption to pioneering companies, privacy, Internet design, and hacker culture. Proposals came in from historians, computer scientists, information scholars, and industry pioneers. At CBI we organized the papers, printed in a privately circulated workshop volume, into four thematic sessions: conceptual foundations, industry foundations, law and privacy, and identity and anonymity. Sessions on Friday, 11 July, were followed by a workshop dinner, with the final session and workshop wrap-up on Saturday, 12 July.

During the workshop sessions, oral presentations were kept brief since all attendees had texts readily at hand in the printed workshop volume. Discussion centered on providing feedback to authors in preparation for publication.

The editorial board of *IEEE Annals of the History of Computing* has approved plans for two special issues to publish revised papers from the event. All papers will go through the journal's standard peer review. CBI Associate Director and past *Annals* Editor in Chief Jeffrey Yost will guest edit the two special issues.

Additional results from CBI's NSF-funded research project include journal articles by co-PI Jeff Yost and graduate-student research assistant Nicholas Lewis that are forthcoming in the *Annals,* the completed oral-history interviews, and the knowledge-networking resources on computer security.[1]

Here is a full list of the papers from the workshop:

- William Aspray (University of Texas), "The Early History of Symantec, 1982–1999"
- James W. Cortada (retired IBM, current Charles Babbage Institute, University of Minnesota), "How an IT Industry Is Born: Is this Happening with IT Security Firms?"
- Laura DeNardis (American University), "The Internet Design Tension between Surveillance and Security"
- Larry Druffel, Rich Pethia, and Bill Scherlis (Software Engineering Institute, Carnegie Mellon University), "The Formation and Operation of CERT: A Retrospective"
- Philip Frana (James Madison University), "Telematics, Transborder Data Flows, and the History of Computer Security"
- Karl Grindal (Cyber Conflict Studies Association), "Artist Collectives versus Hacker Culture: Origins of DDoS"
- Robert E. Johnston, "Information Security History in the Private Sector, 1969–1999"
- Steven B. Lipner (Microsoft), "The Birth and Death of the Orange Book"
- Andrew Meade McGee (University of Virginia), "Privacy as Security: The Deep Political Origins of the Privacy Act of 1974"
- Dongoh Park (Indiana University), "Social Life of PKI: Sociotechnical Development of Korean Public Key Infrastructure"
- Rebecca Slayton (Cornell University), "Automating Judgment: Computer Security Metrics and the Rise of Risk Assessment"
- Michael Warner (US Cyber Command), "Notes on the Evolution of Computer Security Policy in the US Government, 1965–2001"
- Jeffrey R. Yost (Charles Babbage Institute, University of Minnesota), "Access Control Software and the Origin and Early History of the Computer Security Industry"

Reference and Note

1. The completed oral-history interview are available at http://conservancy.umn.edu/handle/11299/59493/browse?type=subject&order=ASC&rpp=20&value=Computer+security, and the knowledge-networking resources on computer security are accessible at https://wiki.umn.edu/CBI_ComputerSecurity/WebHome.

Thomas J. Misa *is the director of the Charles Babbage Institute and the ERA Land Grant Chair in History of Technology in the Department of Electrical and Computer Engineering at the University of Minnesota. Contact him at tmisa@umn.edu.*

SIGCIS at the 2014 SHOT Annual Meeting

The Society of the History of Technology (SHOT) held its 57th annual conference in Dearborn, Michigan,

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

Figure 1. Workshop participants (left to right). Front row: Thomas Misa, Robert Johnston, Karl Grindal, Jeremy Epstein, Bill Scherlis, and Philip Frana. Back row: Rebecca Bace, James Cortada, Jonathan Clemens, Jeffrey Yost, Dongoh Park, Michael Warner, William Vogel, William Hugh Murray, Steven Lipner, Terry Benzel, Andrew Meade McGee, Carl Landwehr, William Aspray, Nathan Ensmenger, Daniel Breck Walker, and Laura DeNardis. Not in photo: Andrew Odlyzko and Rebecca Slayton.

from 6–9 November 2014. This year, the Special Interest Group for Computers, Information, and Society (SIGCIS) held a day-long workshop on "Computing the Big Picture: Situating Information Technology in Broader Historical Narratives." Historian of technology and MIT Professor Jennifer Light gave the plenary talk alongside four workshop sessions and two sessions for works in progress. Light's keynote highlighted the various intellectual genealogies spanning the history of computing, such the history of technology, the history of science, communication and media studies, and architecture and urban planning. She was also quick to observe that, in order to get a broader picture of computing, scholars should attend to the shrinking gap between the popular histories done by journalists, librarians and archivists, and policymakers outside of academic traditions. By considering an array of historical accounts, she urged, new perspectives could cross-pollinate with fields of knowledge that had previously been segregated, creating a much more complex, at times conflicted, "bigger picture" of the history of computing.

Rebecca Slayton chaired the first panel, "Information Technology and the Automated Society." The participants gave individual papers starting with Paul Ceruzzi's "The SHOT/AHA Series on Historical Perspectives on Technology, Culture, and Society: What Should a Booklet on Computing and Information Technologies Contain?" Arvid Nelson presented on "Debates on Automation in the 20th Century: Interpreting New Sources at CBI," Andrew Gansky followed with "The Meaning of Life in the Automated Office," and Ekaterina Babintseva rounded out the panel with "Between Life and Mechanism: The Notion of Information in Warren McCulloch's Theory."

Session Leader Andrew Russell presided over the first work-in-progress session that congregated various social and technological changes in user expectations and use design over the history of computing. William Vogel presented on his paper, "Shifting Attitudes: Women in Computing, 1965–1985." Steven Anderson gave his presentation on "The Digital Imaginary: Mainframe Computers from the Corporate Basement to the Silver Screen, 1946-1968." Margarita Boenig-Liptsin presented on "Making the Citizen of the Information Age: A Comparative Study of Computer Literacy Programs for Children, 1960s–1990s."

This first round of sessions was followed by a lunch sponsored by the IEEE History Committee and related SHOT SIGs to discuss this topic: "Is there a role for the history of technology in the middle school and high school history curriculum?"

"Organizations, Institutions, and Computing" addressed larger systematic impacts that the history of computing has on technological innovations. With Christopher Leslie as acting chair and Cyrus Mody as the session's commentator, Nicholas Lewis gave the first talk, "Computing Behind the Red Line: The HPC History Project at Los Alamos." Chuck House from InnovScapes Institute presented on "The Cisco Heritage Project," and James Lehning followed with "Technological Inovation and Commercialization: The University of Utah Computer Science Department, 1965–1975." Michael Castelle concluded with "Making Markets Durable: Transaction Processing in Finance and Commerce."

With the widening interest and attention paid to game studies and user design in academia, "At the Interfaces: Users and Games" chaired by Gerard Alberts provided refreshing perspectives on gender, museum education, and power relations. Kimon Keramidas' "The Interface Experience" introduced a museum exhibition that would also provide a historical account of personal computing. Katherine McFadden presented "Hand Sewn Computing: Women's Hobbies, Needlework, and Computer Electronics." Jonathan Scott Clemens followed with "The Most Blatant Testimony We Have to American Waste:

Moral Panic and Video Arcade Games, 1978–1983." Michael McGovern finished the session with a historical reframing of usership with "Reframing Power Relations in the Historiography of Computing: Examples from Early Medical Genetics and Calculator User Groups."

For the second work-in-progress panel, Jason Gallo led the session with questions about the application of disparate histories of computing, from third-world engagements to implications of an early American computer education, from the broadening applicability of IT workforces to aesthetic and programmatic legacies in information visualization. Accordingly, Beatrice Choi presented on "Ser Técnico: Localized Technology Transfer, Emerging Technical Actors, and the Brazilian Computer Industry." William Aspray followed up with "How to Frame a Study of the History of IT Education and its Relation to Broadening Participation in the IT Workforce in the United States," which served as a discussion on his latest book and ideas. Alex Campolo presented on "White-Collar Foragers: Ecology, Economics, and Logics of Information Visualization."

Nathan Ensmenger presided over the last panel, "Designing and Making Computers." William McMillan presented on "Technical Trends in the History of Operating Systems." Lav Varshney gave a talk on "Block Diagrams in Information Theory: Drawing Things Closed." Barbara Walker followed with "Gossip, Storytelling, and the Spread of Innovation: The Von Neumann and Lebedev Computer Projects in Comparison." Finally, Gerardo Con Diaz closed this session with "Embodied Software: Patents and Software Development, 1946–1970." By focusing on the histories of various aspects and features of computing such as operating systems, information theory applied through diagrams, the socialization of computing via competition, and the patent strategy of "embodying software," this panel approached the "bigger picture" history of computing by considering its overlooked computing parts.

Beatrice Choi *is a PhD student and teaching instructor in the Rhetoric and Public Culture program at Northwestern University. Contact her at beatrice.choi@u.northwestern.edu.*

The 2014 SHOT conference included a SIGCIS workshop on "Computing the Big Picture: Situating Information Technology in Broader Historical Narratives."

History and Philosophy of Computing: New Seminars in France

Two new seminars are being set up in Paris and in Lille on the history and philosophy of computing. They aim at developing interdisciplinary approaches to computing, in the wake of the History and Philosophy of Programming (HaPoP) and History and Philosophy of Computing (HaPoC) conferences to which the organizers participated in the past three years in Ghent, Birmingham, and Paris (see http://hapoc2013.sciencesconf.org).

"History and philosophy of computing: practices, concepts, methods" is a monthly seminar, organized at the Paris Institute for the History and Philosophy of Science and Technology (IHPST) by two philosophers and a historian:

- Baptiste Mélès (CNRS, Archives Poincaré at the University of Nancy)
- Maël Pégny (CNRS, University of Paris 1, IHPST)
- Pierre Mounier-Kuhn (CNRS and University of Paris-Sorbonne)

Starting in January 2015, the program features the following lectures:

- Edgar G. Daylight (University of Utrecht), "From the Pluralistic Past to the Pluralistic Present in Programming"
- Mark Priestley, "Making a Place for Programmers: A New Look at John von Neumann's 'First Program'"
- Giuseppe Primiero (Middlesex University London), "Computer Science as Abstraction and Control"
- Frank Varenne (University of Rouen), "Computing between Emulation and Simulation"

Gérard Berry (Collège de France) and Gilles Dowek (INRIA) are registered for two more sessions on computability.

In partnership with the former, a seminar on the "Interactions between logic, language and computing: History and philosophy" has been organized in Lille by Liesbeth De Mol (CNRS/STL, University of Lille 3), Alberto Naibo (IHPST, University of Paris 1, ENS), Shahid Rahman (STL, University of Lille 3), and Mark van Atten (CNRS/SND, University of Paris-Sorbonne). The 2015 program includes these lectures:

- Giuseppe Primiero (Middlesex University), "Software Theory Change"
- Yonathan Ginzburg (University of Paris VII), "Quotation, Diagonalization, and Dialogue"
- Pierre Mounier-Kuhn (CNRS & University of Paris-Sorbonne), "Logic and Computer Science: A Filiation or Encounter Process?"
- Amirouche Moktefi (Tallinn University of Technology), "A Boolean Legacy: The Problem of Elimination in the Algebra of Logic"
- Baptiste Mélès (CNRS, Archives Henri Poincaré, University of Nancy), "Digital Networks and Dialogical Logic"
- Jacqueline Léon (CNRS, HTL, Laboratoire d'histoire des théories linguistiques, University of Paris VII), "The Controversy between Yehoshua Bar-Hillel and Margaret Masterman over Language Formalization and Machine Translation (1958–1960)"
- Edgar Daylight (Utrecht University), "Towards a Dutch Perspective on the Beginnings of Machine-Independent Programming"
- Mark Priestley (UCL), "'Visualizing Computation': From the Differential Analyzer to the Flow Diagram, By Way of Some Little Known ENIAC Drawings"
- Maarten Bullynck (University of Paris VIII), "Excavating the Roots of the Chomsky-Hierarchy: Computational and Linguistic Practices at MIT before 1963"

This is in addition to the 20-year-old seminar on the history of computing, created at the Conservatoire National des Arts & Metiers (CNAM, Paris) shortly after the second conference on the history of computing in France. Over the years, this seminar has moved to the Sorbonne University and back to CNAM in 2011. It has now three "anchors":

- François Anceau, a specialist of computer architecture and (retired) professor at CNAM and at the Grenoble University;
- Isabelle Astic, curator at Musee des Arts & Metiers, in charge of computing and network collections; and
- Pierre Mounier-Kuhn, historian at CNRS and University of Paris-Sorbonne.

The principle of the seminar is to combine or alternate historians' and social scientist's approaches with actors' testimonies (see www.musee-informatique-numerique.fr). For instance, last year the program included lectures by

- Christophe Lecuyer, "Moore's Law and the Governance of Innovation";
- Marie-Aline de Rocquigny, "The Emergence of Managerial Informatics. Actors and Representations (1960s–1970s)";
- Pierre Mounier-Kuhn, "Golden Age Lost? Women's Careers in Computer Science in France (1955–1980)";
- Marie d'Udekem-Gevers, "The Mathematical Machine of IRSIA-FNRS: An Unknown Episode of the History of Computing in Belgium";
- Rémi Després, "Once upon a Time before the Internet: The Worldwide X.25 Network";
- Louis Pouzin, "Cyclades. A Contribution of French Research to the Development of the Internet";
- François Anceau, "The Parallelizing of Sequential Program Execution: From Tomasulo to Xeon"; and
- Jean-Jacques Quisquater, "On the Origins of RSA Cryptography, From Fermat to the Smart Card: Whose Invention? Surprises in Store."

This last presentation is part of the preparation of the exhibition celebrating the 40th anniversary of the smart card at the Musee des Arts & Metiers (June 2015).

Pierre Mounier-Kuhn *is a CNRS researcher at the University of Paris-Sorbonne. Contact him at mounier@msh-paris.fr.*

Reviews

Andrew Russell, Editor

Thomas J. Misa, *Digital State: The Story of Minnesota's Computing Industry*, University of Minnesota Press, 2013.

Simply put, Thomas Misa's 2013 *Digital State* should be an addition to the "must read" lists of serious students and practitioners of the history of computing. In its 300 dense, but readable pages, Misa achieves several noteworthy goals. Foremost, he presents a powerful argument for the importance of Minnesota within the history of computing and the computer industry and of the US military as its critical, enabling sponsor and customer across this history. With this book, he continues his valuable contribution to the discussion of place in the history of technology (as in his recent article on transnational history in *History and Technology* with Johan Schot[1]) and to the importance of the state and military in seminal technological developments (as in his 1985 chapter on the transistor in Merrit Roe Smith's edited volume[2]). As Misa details, some of the most storied names in the history of the computing industry—Engineering Research Associates (ERA), Control Data Corporation, Sperry Rand, and Honeywell—were centered in the Twin Cities region of Minnesota, and all used R&D contracts and lead orders from the military to build digital computing and commercial markets for it during the Cold War.

Another goal accomplished by Misa in *Digital State* is the creation of an accessible general history of modern computing, told through the lens of the major developments in Minnesota. For example, the establishment of ERA in St. Paul is key to both the launch of the digital computer industry in general and the Minnesota industry in particular. Rather than sticking strictly to the details of why this spinout of an engineering group from the Navy's cryptologic organization took root in Minnesota, Misa situates this within a crisp, thorough, and insightful overview of the simultaneous emergence of machine methods in cryptanalysis and electronic digital computing during World War II, along with their intersections. The narrative winds from NCR's creation of electromechanical bombes for Allied attacks on the infamous Enigma machine through the creation of the ENIAC computer in Philadelphia and on to the earliest magnetic-drum-equipped mainframes made by ERA.

In this and other episodes, Misa's assumes little about his readers' background exposure to the history of computing, which lets the book stand on its own and makes it suitable as an insightful overview text for a graduate seminar or advanced undergraduate course grappling with the history of computing or, perhaps, the training of electrical engineers and computer scientists. The definitional role of the US military in the early computer industry is exemplified in the startup story of ERA. In that case, machines first delivered to the US National Security Agency and its predecessor organizations were then introduced as commercial products for the laboratory and the office. Absorbed into Remington Rand in 1952 and thus into Sperry Rand in 1955, the trajectory of ERA into the period of Big Iron exemplifies the themes of consolidation and competition in the rise of the mainframe. Sperry Rand's production of mainframes in the Twin Cities did much to establish the region in the industry, eventually employing more than 10,000.

Honeywell, the venerable maker of industrial controls and thermostats, as well as one of the region's largest employers, jumped into digital computing in the 1950s as well—first in joint activities with Raytheon and then, in 1960, with a mainframe division of its own. In the mid-1960s, Honeywell acquired a leading Massachusetts maker of minicomputers, Computer Control, and in 1970, acquired General Electric's computer operations, including its line of innovative timesharing systems. Combining these capabilities, Honeywell created a national network of modem and terminal accessible computers, available as a time-sharing service. Its local rival, Control Data Corporation, did the same, and the CDC Cyber Net network was at the time more extensive than the ARPANET.

Control Data was the successful second startup by key engineers involved in ERA. Founded in 1957, Control Data again aimed at the US military market, this time in the demand for supercomputers by nuclear weapons laboratories, most prominently Lawrence Livermore, and by the NSA. Other users in science and engineering soon followed. Around the time of Control Data's formation, IBM moved into southern Minnesota, creating a major manufacturing facility on 400 acres near Rochester. Soon the modernist Saarinen building was home to IBM's production of minicomputers and disk drives as well as R&D activities. Eventually, IBM Rochester moved into supercomputing as well, with the famous Blue Gene line of computers, and key subsystems for the *Jeopardy*!-winning Watson emerged from its factory floor.

In treating these developments, Misa effectively redeploys the approach Phil Scranton uses for US specialty manufacture[3] to discuss the digital computer industry. In particular, Misa finds Scranton's concept of an "industrial district," with lead "integrated anchor" firms and "specialist auxiliary" companies, to fit the Minnesota story well. He makes good use of Scranton's

1058-6180/15/$31.00

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page

Hopefully Misa's success with *Digital State* will build momentum that encourages additional studies of localized productive networks and industrial districts.

concept and insights in elaborating the connections between computer makers and suppliers like 3M and local precision manufacturers and machine shops.

Finally, Misa accomplishes a goal that is simultaneously a remedy and, hopefully, an inspiration. In his story of computing in Minnesota, he ably demonstrates that many of the practices and dynamics that are too often hailed as distinctive of Silicon Valley were present at the same time, and sometimes earlier, in the Twin Cities and their surroundings. Silicon Valley and the Twin Cities (and one might also add the Boston region) were host to the dynamics of start-ups, spinoffs, and venture capital organizations created by successful engineer-entrepreneurs, government contracts, military aerospace, and intelligence markets as platforms for commercial developments and historically important innovations. *Digital State* shows that localized production networks confronted both common and specific contexts in the development of electronic digital computing as well as the importance of the differences between the responses of these localized networks and the transactions, shifts, and competitions among them.

Of course, natural pairings exist between *Digital State* and Paul Ceruzzi's study of the Dulles Airport corridor in the DC region, Christophe Lécuyer's *Making Silicon Valley*, and AnnaLee Saxenian's *Regional Advantage*.[4] Hopefully Misa's success with *Digital State* will build momentum that encourages additional studies of localized productive networks and industrial districts from which we may begin to more fully see how the intranational and international connections between these localized networks simultaneously produce national and transnational histories of computing.

References

1. T.J. Misa and J. Schot, "Inventing Europe: Technology and the Hidden Integration of Europe," *History and Technology*, vol. 21, no. 1, 2005, pp. 1–19; doi: 10.1080/07341510500037487.
2. T.J. Misa, "Military Needs, Commercial Realities, and the Development of the Transistor, 1948–1958," *Military Enterprise and Technological Change*, M.R. Smith, ed., MIT Press, 1985, p. 253–287.
3. P. Scranton, *Endless Novelty: Specialty Production and American Industrialization, 1865–1925*, Princeton Univ. Press, 2000.
4. P. Ceruzzi, *Internet Alley: High Technology in Tysons Corner, 1945–2005*, MIT Press, 2008; C. Lécuyer, *Making Silicon Valley: Innovation and the Growth of High Tech, 1930–1970*, MIT Press, 2005; A. Saxenian, *Regional Advantage: Culture and Competition in Silicon Valley and Route 128*, Harvard Univ. Press, 1996.

David C. Brock *is a senior research fellow and coauthor of* Makers of the Microchip: A Documentary History of Fairchild *Semiconductor (MIT Press, 2010). Contact him at* dcb@dcbrock.net.

Raiford Guins, *Game After: A Cultural Study of Video Game Afterlife*, MIT Press, 2014.

In *Game After*, Raiford Guins sets out to evaluate the place of videogames as historical objects from a museum preservation and material culture perspective. Guins' reexamination of the afterlife of videogames provides a new contextualization of arcade machines and game consoles as relics of an earlier gaming era. Furthermore, it invigorates the field of game studies with the professional historicity it has been lacking. The book's braiding of public history and traditional scholarship produces a down-to-earth odyssey of discovery and reflection, as Guins leads the reader through the various archives, museums, collections, and professional workshops where preservationists and engineers collaborate to keep failing electronic parts and time-worn wooden cabinets from reaching true death. Barring the occasional obscure reference, Guins writes with an authoritative voice, juxtaposing literary and historical theory with real-world examples, rarely forcing the reader to question his conclusions. Guins is quick to note in his introduction that the current state of game studies is somewhat lacking from a historiographical perspective, and although his book is specific in its scope, it provides a solid example of how to include useful historiographical methods (most importantly, material culture) with game studies while keeping it readable.

Guins does not focus on understanding old videogames as they were in the time of their origin, as if put in a time capsule. Instead, he emphasizes their post-death state: when an arcade game machine stopped being popular or when Atari cartridges are found more often in a landfill than in secondhand stores, for example. In other words, Guins writes "about the historical life cycles of video games and the diverse ways we experience them today" (p. 4). The various preservation efforts he interrogates shed a new light on the popularity of emulators and the current trends of retro-gaming. The significance of Guins' approach lies in the imminent material death of decades old videogames and the unique requirements needed to maintain old gaming hardware. These requirements become more obvious with each chapter, as Guins makes his way through a plethora of recent preservation endeavors, exemplifying the nonuniform and place-specific needs and practices of preservation-restoration efforts. Where certain museums accept the hardware death of their exhibition, augmenting visitor experience with modern emulation programs, restoration specialists (such as the Vintage Arcade Superstore) possess large collections of disassembled parts from broken machines repurposed to repair and restore arcade games for the personal collector (see Chapters 1 and 6).

As a result of the book's scope, its chapters seem disconnected at times, as Guins jumps from museums to arcades to discussions of box art. Of particular interest to public historians and museum studies scholars, the first chapter discusses the place of videogames in museums. Guins compares the different approaches of several videogames exhibition practices of museums, such as the Strong Museum of Play's International Center for the History of Electronic Games and the Computer History Museum. He elaborates on the difficulties of achieving a balance between providing visitors with a tactile experience, while preserving consoles and arcades from further wear. Chapter 2 delves into the archival collection of videogames, where Guins discusses the difficulties of videogame archiving. Although the chapter's theoretical framework is important as a general argument about the place of archives in academe, it has the secondary usefulness of listing several key videogame archives in the United States. The third chapter is a lengthy and in-depth survey of recent arcade projects, illustrating the various ways museums and enthusiasts re-create the feel of the arcade in a combination of nostalgia and history. In Chapter 4, he analyzes the design and marketing choices put into the creation of Atari games box art. This analysis illustrates the usefulness of material culture studies in understanding marketing trends in the game industry and the creation of brand recognition. Guins revisits the mythical Atari Alamogordo landfill in the fifth chapter. Here, Guins not only focuses on the landfill legend, but he also delves into the archeological and historical meaning of games as "trash" and analyzes what made *E.T.* a bad game. Guins' inquiry puts the development of the game created by Howard Scott Warshaw in much needed perspective, showcasing the time restrictions and corporate pressures to release the game within an inadequate amount of time.

Although the book relies too heavily on theory for the lay videogame enthusiast, historians of computing may find it useful because Guins provides a unique mixture of method and theory that can be used in other aspects of hardware and software history. Public historians and those interested in museum studies and preservation, especially of recent technology, will also find much of interest in this book. For scholars interested in game studies or the history of videogames, this book is a must have.

Itamar Friedman *is a history of science and technology PhD candidate in the Department of History at the University of South Carolina. Contact him at friedmai@email.sc.edu.*

Hallam Stevens, *Life Out of Sequence: A Data-Driven History of Bioinformatics*, The University of Chicago Press, 2013.

Hallam Stevens' *Life Out of Sequence* opens with the description of a contemporary biology laboratory in which "[i]n half of the lab, benches had been replaced by long rows of desks, each holding three or four computers" (p. 5). The transformation of the physical space and the (partial) replacement of the traditional biologists' tools with hardware and software is an apt image for summarizing the change brought in the study of life by bioinformatics. This interdisciplinary field has grown as a result of contributions from biology, computer science, mathematics, statistics, and software engineering, just to mention the main contributing disciplines.

Life Out of Sequence pursues the investigation of bioinformatics relying on a joint historical and ethnographic approach. To build his narrative, Stevens makes use of archival research, fieldwork in bioinformatics ventures, and more than 70 interviews with people engaged in the development of the discipline. As the subtitle makes clear, it is the role of data, mainly gathered by sequencing DNA, RNA, and proteins, that is the driving force behind this account of bioinformatics. Following the process that transforms "wet" biological samples into data that can be archived in a database and shared worldwide via the Internet, Stevens describes in six chapters "how biological work and biological knowledge is shaped and textured by digital infrastructures" (p. 203).

The book begins with an historical account of the development of computers during and after World War II and their encounter with biology. It examines early computer projects, such as MOLGEN, which is aimed at combining expertise in molecular biology with techniques from artificial intelligence, and describes unusual careers like the one of the physicist Walter Goad, who was instrumental in the establishment of the National Institutes of Health (NIH) database of DNA sequences (GenBank), and the pioneering work in bioinformatics of the biologist James Ostell. Chapter 2 focuses on the new (but not uncontested) approach to making biological knowledge permitted by the use of computers and reflects on the challenges posed by data-driven research.

> **Stevens raises questions about the biological, medical, and social implications of a biology driven by sequencing data and speculates about a future "end"of bioinformatics.**

Chapters 3 and 4 rely more directly on Stevens' fieldwork in biological laboratories in the United States and Europe and describe the physical and virtual spaces in which bioinformatics is enacted. The author guides the reader in the business-like environment of the Broad Institute headquarters in Cambridge, Massachusetts, "eight stories of shimmering glass and metal" (p. 79), and in the factory-like establishment of its sequencing center (Chapter 3). He also offers an insight into the computer networks and the standards that have been devised to share sequencing data (Chapter 4).

Chapter 5 returns to an historical perspective to show how databases have influenced and constrained the practices of biology, using as case studies Margaret Dayhoff's *Atlas of Protein Sequence and Structure*[1] and the already mentioned GenBank (operational since 1982). The final chapter of the book shifts the focus from databases to the visualization strategies adopted for facilitating the use of sequencing data. A major concern of the author is discussing how these visualization tools shape and constrain biologists' understanding of data.

In conclusion, Stevens raises questions about the biological, medical, and social implications of a biology driven by sequencing data and speculates about a future "end" of bioinformatics, an end not due to a refusal of digital tools but justified on the contrary by the ubiquity of bioinformatics practices. Stevens' interviewees who consider bioinformatics "of marginal importance to 'real' biology" or "as the same old biology dressed up in computer language" (p. 43) would rather disagree with this vision.

The book is engaging and detailed, but readers not already familiar with the sequencing process or the digital tools discussed would have benefited from a glossary of biological and computing terms. A debatable point in Stevens' account is the great emphasis that he places on physics in the development of bioinformatics. All the other disciplines that contributed to the growth of this interdisciplinary venture are somehow belittled by his choice. This is particularly evident in relation to computer science and computational science, which were already established by the early 1990s when Stevens sets the establishment of bioinformatics. Yet although technical aspects, such as ontologies, are discussed at length in the book, there is not much about the expectations and agendas that brought computer scientists and computational scientists into biology. However, such computer experts must have been (and still are) some of the main characters behind this story, alongside biologists; after all, the hybrid discipline developed is named *bioinformatics* and is considered something different from the fields of biophysics and biostatistics that also make use of digital tools for computing and data management.

That said, *Life Out of Sequence* deserves credit as an ambitious account of an interdisciplinary scientific enterprise. It is a stimulating book for all the readers interested in the intertwining of computers and biology and in the history of data and their management.

Reference

1. M. Dayhoff, *Atlas of Protein Sequence and Structure*, Nat'l Biomedical Research Foundation, 1966.

Giuditta Parolini *is a postdoctoral fellow in the Institut für Philosophie, Literatur-, Wissenschafts- und Technikgeschichte at the Technische Universität Berlin. Contact her at giudittaparolini@gmail.com.*

M. Deguerry and R. David, *De la logique câblée au calculateur industriel. Une aventure du Laboratoire d'Automatique de Grenoble*, Eda Publishing, 2008.

The issue of research-industry collaboration became an obsession in the 1960s for many European planners, who compared the barriers to innovation they perceived in their countries with the American model. Partly for this reason, success stories have been rare in the French computer industry. This book narrates one of them, a case of fruitful technology transfer from a university to a private company during the heyday of the minicomputer.

An electrical engineer from the *Institut Polytechnique de Grenoble* (Grenoble Polytechnic Institute), René Perret had devoted his doctoral dissertation to the regulation of large electricity transport networks and had spent six months at Howard Aiken's computing laboratory in Harvard and at the National Bureau of Standards in Washington, DC. Perret understood the need for expertise and training in emerging fields like regulation, servo-mechanisms, and automatic control. In 1958 he created a servo-mechanism team within the university's Applied Mathematics Laboratory and then in 1962 established his own Laboratoire d'Automatique de Grenoble (LAG)—an acronym that could not be misinterpreted in French. Perret maintained close relationships with international figures in this field, such as Winfrid Oppelt from Darmstadt, or Yasundo Takahashi from Berkeley who came as visiting professor to the University of Grenoble.

The development of the laboratory for automatic control followed the same process model as the one observed in computer science.[1] It attracted students, who eventually completed doctorates while working as assistants and could undertake applied research projects. Its resources came from contracts with private industry, nationalized companies, and government agencies as well as from the Ministry of Education. This generated revenue and new research topics as well as practical experience in cooperating with firms.

Among various industrial partners, the Battelle Institute in Geneva employed Perret as a consulting engineer and entrusted his laboratory with the design of a calculator, Alpac (Analogue & Logic Process Automation Computer).

In Grenoble the Mors company, a venerable automobile maker that had converted to manufacturing railway signals and automatic systems, established a joint research team with LAG in the Polytechnic Institute. This team was sponsored by DGRST, a new government agency that had recently been set up specifically to support research-industry collaborations. The first outcome was a family of digital modules designed at LAG, using resistor-transistor logic (RTL) and marketed as Logimors.

In 1963, two of Perret's doctoral students were assigned the task of designing a small industrial computer under contract with Mors. The book vividly describes the project and its constraints, the growing research team, the technological choices, the experience gained with laboratory mock-ups, and the many efforts still needed to transform the latter into a rugged machine able to work in chemical or mechanical plants. With all due qualifications, the MAT01 could boast being the first computer based on ICs developed and marketed in France, perhaps even in continental Europe. Mors

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

presented it in October 1965 and, over the next three years, sold a modest 20 units, gaining 15 percent of the French market for this class of machines, in competition with DEC PDP-8 and other computers from various French makers. The authors devote a useful chapter to the first clients—for example, the MAT01 processed data for the stress tests of the Concorde supersonic airliner and was used to control the regional power grid in Naples, Italy. The MAT01 was marketed in 1965, and its designers defended their doctoral theses six months later. Accordingly, the authors remark that it was a rare example of a negative time lag between invention and innovation!

Because demand grew faster than Mors' investment capability, the computer and automation division was sold to another established firm in the same field, Télémécanique. Télémécanique used the MAT01 to diversify its offer and then developed a whole family of minicomputers based on TTL technology. Their commercial success contrasted with the ups and downs of the government-sponsored Plan Calcul, at the same time. It required a new factory that employed 700 people in the early 1970s.

To many readers throughout the world, this is a rather common story of a technical development in a university department of electrical engineering, leading to a transfer of technology and talents to a private company that converted it in a commercial success. Yet the book is particularly welcome because the literature on European laboratories devoted to automatic control is scarce compared with the vast historiography on computer science, and little has been published on European minicomputers. Moreover, to social scientists it may serve as a good case study in revisionist history. This story challenges the established view of a centralized France where not much happened outside Paris, of French scientists living in an ivory tower and not involving themselves with technology, and of conservative, if not Malthusian, managers in French small businesses.

Yet the innovative configuration in Grenoble was seriously impacted a few years later by decisions from Paris. In 1975, following the termination of the Plan Calcul, Thomson-CSF, the large, diversified electronics conglomerate with many connections with the civilian and military administrations, maneuvered to take control of the minicomputer business of Télémécanique as well as of CII. The outcome was predictable: Thomson merged the two subsidiaries, starting years of conflicts between rival teams and product lines, a succession of managers, and a slowdown in growth, while other minicomputer vendors increased their share of the European market.

Meanwhile, LAG went on as one of France's main laboratories in its field and still exists as GIPSA-Lab. It celebrated its 50th anniversary in 2008, with the publication of this book.

The authors themselves were key actors in the story. Their later interest in history benefitted from the resources of Grenoble-based Aconit, one of the most active computer heritage societies in Europe.[2]

Well-documented and illustrated, the book ends with useful appendices, including a list of doctoral dissertations prepared at LAG, that shed light on the evolution of the field. Overall, it does a good job in weaving three threads together:

- a monograph of a laboratory with a eulogy of its founding father;
- the saga of a series of machines and of the firms that manufactured and sold them; and
- an engineer's introduction to the history of technologies: successive approaches to control and command, components, systems, and the mathematics of technology.

The introduction serves to bring a deeper sense of scientific and historical culture to younger generations. The authors' success with these threads, and their passionate desire to revive the excitement of their adventure, make the book interesting for historians of higher education, technology, and the computer industry.

References and Notes

1. For more on this model and the Grenoble context, see P. Mounier-Kuhn, "Computer Science in French Universities: Early Entrants and Latecomers," *Information & Culture*, vol. 47, no. 4, 2012, pp. 414–456.
2. See http://aconit.org/spip/spip.php?page=sommaire-eng for more details on the *Association pour un conservatoire de l'informatique et de la télématique* [Society for the conservation of computing and digital networks heritage].

Pierre Mounier-Kuhn *is a CNRS researcher at the University of Paris-Sorbonne. Contact him at mounier@msh-paris.fr.*

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

continued from p. 96

company's 25th anniversary alongside other Canon products such as the SureShot automatic camera and PalmPrinter calculator. "In addition to general consumer items," the ad explained, "Canon's optic, electronic and engineering capabilities contribute to industrial productivity, as well as professional and humanitarian aims."[13] Technical, economic, and social factors all contributed in part to the production and distribution of the Communicator.

Similarly, TI explicitly marketed the Vocaid as a dedicated AAC device and as a member of the company's product family. The Vocaid Owner's Manual links the communication aid to TI's corporate lore, noting that "Texas Instruments invented the integrated circuit, the microprocessor, and the microcomputer, which have made TI synonymous with reliability, affordability and compactness. Vocaid carries on TI's tradition of technology leadership."[14] Launched in 1982, the Vocaid was a direct spinoff of a preexisting TI product, the Touch & Tell toy, introduced a year earlier. In both products, printed interchangeable panels overlaid a touch pad that, when pressed, activated an electronic circuit to verbally pronounce sounds, letters, numbers, words, and phrases. A 1983 report from the US Congress' Office of Technology Assessment noted that the Vocaid "might well have never been modified and commercialized had Texas Instruments not already had a running start on this technology."

The Touch & Tell was based on TI's prior innovations in synthetic speech and solid-state memory. In 1978, the TI Speak & Spell toy became the first consumer electronic device to duplicate the human vocal tract on a single chip of silicon.[15] The aforementioned *Newsday* article highlighted this repurposing, noting of the Vocaid, "Now, the same technology that produced the little child's toy that E.T. the extra-terrestrial rearranged to 'phone home' with is also being used to give non-verbal people a new way to find a voice."[1] Interestingly, it is the Speak & Spell's own reinvention in the 1982 film E.T. that catapulted it into the cultural zeitgeist.[16]

Ultimately, the Vocaid was, by most accounts, a failure. Former Vocaid engineer Paul Michaelis commented, "It just sat on the shelves. It was a tremendous disappointment.... Nobody bought it."[17] While the Touch & Tell sold for approximately $40, the Vocaid was priced much higher at $150, beyond most individual's means.[1] Various policy and economic disincentives led to the Vocaid's eventual discontinuance, including delayed and partial reimbursement from third-party funders such as Medicare, high marketing costs, and difficulties in identifying prospective users because of the diverse range of disabilities that manifest in an inability to talk.[12] Arlene Kraat, director of the CUNY-Queens College Augmentative Communication Center, noted her frustration over barriers to full utilization. Devices like the Vocaid were "capable of a lot," she said, "but we just can't get it to the people who need it. There's a lot of money to develop talking cars, talking toys and talking refrigerators because there's a better profit in it."[1]

> **The iPad, when used for AAC, carries with it the legacy of struggles over private and public funding for assistive technologies.**

Recovering Disability

Repurposing, such as that of the Speak & Spell, Touch & Tell, and Vocaid, has a long history. Various historians of communication technologies have chronicled how moments of "progress" and "innovation" always enter into preexisting systems[18] and how technological "failures" produce new forms of knowledge.[19] Media studies scholars Jay Bolter and Richard Grusin use the term "remediation" to describe the refashioning of new media out of prior media forms.[20] With respect to the history of assistive devices, Mara Mills has written, for example, about how throughout the 20th century, AT&T and Bell Labs drew on notions of "normal" hearing to develop telephone technologies and then transformed telephone parts into components for hearing aids and audiometers to measure hearing loss.[21] At the dawn of the PC era, AAC devices remediated advancements in microcomputing, and the Vocaid was later appropriated by hackers and hobbyists engaged in the art of circuit bending (the

IEEE Annals of the History of Computing | Previous Page | Contents | Zoom in | Zoom out | Front Cover | Search Issue | Next Page | Qmags THE WORLD'S NEWSSTAND®

The history of AAC sheds light on the inexorable, but understudied links between the history of communication technologies and disability history.

creative rewiring of electronic devices and toys to create new and unusual sounds.)[22]

The iPad, as an AAC device, also remediates the Vocaid and the Touch & Tell. Instead of printed panels, the iPad supports various apps. The tablet can be an assistive technology, a videogame console, and myriad other tools. However, the iPad, when used for AAC, also carries with it the legacy of struggles over private and public funding for assistive technologies. Although most apps available in iTunes are free or cheap, AAC apps are some of the most expensive in the Apple marketplace, costing $200 to $300. Meanwhile, state agencies and insurance companies generally do not cover the iPad (which currently sells for $299 to $929) because it is not considered "durable medical equipment," meaning a technology exclusively dedicated to AAC.[23]

The comparative case study of the Vocaid and the iPad illustrates how the research, development, commercialization, use, and reuse of augmentative and alternative communication devices is embedded within the history of mobile computing. The Vocaid, Touch & Tell, and Speak & Spell not only carry on "TI's tradition of technology" but also join devices such as portable televisions and radios in an extended lineage of mobile communication technologies.[24] Moreover, the history of AAC sheds light on the inexorable, but understudied links between the history of communication technologies and disability history. Disability and impairment are not simply addressed by rehabilitation and recovery through assistive technology; individuals with various disabilities need to be recovered from and rewritten into the history of how communication technologies are designed, marketed, and adopted. While "new machines give the silent a chance to speak," the history of communication technology should listen more closely to the economic, social, and cultural logics that whisper through today's speaking machines.

Acknowledgments

Many thanks to Nathan Ensmenger and Caroline Jack for their comments and suggestions.

References and Notes

1. M. Unger, "New Machines Give the Silent a Chance to Speak," *Newsday*, 7 Mar. 1983, p. A8.
2. J. De Avila, "Donor of the Day: Giving the Gift of Voice With the Touch of a Finger," *Wall Street J.*, online, 24 Dec. 2011; http://online.wsj.com/news/articles/SB10001424052970204464404577116603396144464.
3. For an example of techno-utopianism, see N. Negroponte, *Being Digital*, Vintage Books, 1996.
4. G. Goggin and C. Newell, *Digital Disability: The Social Construction of Disability in New Media*, Rowman & Littlefield, 2003.
5. N. Couldry, *Why Voice Matters: Culture and Politics After Neoliberalism*, Sage, 2010.
6. See E. Ellcessor, "Bridging Disability Divides," *Information, Communication & Society*, vol. 13, no. 3, 2010, pp. 289–308; Goggin and Newell, *Digital Disability*; M. Mills, "Deafening: Noise and the Engineering of Communication in the Telephone System," *Grey Room*, vol. 43, 2011, pp. 118–143; J. Sterne, *The Audible Past: Cultural Origins of Sound Reproduction*, Duke Univ. Press, 2003.
7. D. Beukelman and P. Mirenda, *Augmentative and Alternative Communication: Supporting Children and Adults with Complex Communication Needs*, 4th ed., Brookes Publishing, 2013.
8. G. Vanderheiden, "Providing a Child with a Means to Indicate," *Non-Vocal Communication Techniques and Aids for the Severely Physically Handicapped*, G. Vanderheiden and K. Grilley, eds, University Park Press, 1976, pp. 20–76.
9. L. Stone, "Therapy Penetrates Silence of Autistic," *Globe and Mail*, 23 Apr. 1981, p. T1; Vanderheiden, "Providing a Child with a Means to Indicate."
10. "Electronic Speaking," *IEEE Professional Comm. Soc. Newsletter*, vol. 21, no. 2, Apr. 1978, p. 12.
11. M. Williams, "Transitions and Transformations," *Proc. 9th Ann. Minspeak Conf.*, 1995, p. 3.
12. US Congress, Office of Technology Assessment, *Health Technology Case Study 26: Assistive Devices for Severe Speech Impairments*, Dec. 1983.
13. "A Message From Canon on its 25th Anniversary," *Los Angeles Times*, 7 Oct. 1980, p. E18.
14. Texas Instruments, *Vocaid Owner's Guide*, p. 10.

15. Texas Instruments, "TI Talking Learning Aid Sets Pace for Innovative CES Introductions," 11 Jun. 1978; www.ti.com/corp/docs/company/history/timeline/eps/1970/docs/78ces-vignette.htm.
16. Although it is beyond the scope of this article, the role of electronic toys is also largely underexplored within the history of personal computing and miniaturization.
17. "Emergency Access Advisory Committee Meeting," YouTube video, 20 June 2011; https://www.youtube.com/watch?v=KSF4ekob3oE&list=UUIWI0h4WenWWPlJrqU7GGsg.
18. For example, see C. Marvin, *When Old Technologies Were New: Thinking About Electric Communication in the Late Nineteenth Century*, Oxford Univ. Press, 1988; D. Edgerton, *The Shock of the Old: Technology and Global History Since* 1900, Oxford Univ. Press, 2006.
19. K. Lipartito, "Picturephone and The Information Age: The Social Meaning of Failure," *Technology and Culture*, vol. 44, no. 1, 2003, pp. 50–81.
20. J. Bolter and R. Grusin, *Remediation: Understanding New Media*, MIT Press, 2000.
21. Mills, "Deafening: Noise and the Engineering of Communication in the Telephone System;" M. Mills, "Hearing Aids and the History of Electronics Miniaturization," *IEEE Annals of the History of Computing*, vol. 33, no. 2, 2011, pp. 24–45.
22. M. Mirapaul, "Circuit Benders Unlock the Long Riffs in Short-Circuits," *New York Times*, 8 Apr. 2004, p. G4.
23. S. White and J. McCarty, "Bottom Line: Reimbursement for AAC Devices," *The ASHA Leader*, 11 Oct. 2011; www.asha.org/Publications/leader/2011/111011/Bottom-Line–Reimbursement-for-AAC-Devices.htm.
24. See M. B. Schiffer, *The Portable Radio in American Life*, Univ. of Arizona Press, 1991; L. Spigel, "Media Homes: Then and Now," *Int'l J. Cultural Studies*, vol. 4, no. 4, 2001, pp. 385–411.

Meryl Alper *is a doctoral candidate in communication at the Annenberg School for Communication and Journalism at the University of Southern California. She is currently completing a dissertation that explores the relationship between disability, mobile devices, and family media use. Contact her at malper@usc.edu.*

Augmentative, Alternative, and Assistive: Reimagining the History of Mobile Computing and Disability

Meryl Alper
University of Southern California

Editor: Nathan Ensmenger

"New machines give the silent a chance to speak," read the headline of an article in *Newsday* profiling the Nassau United Cerebral Palsy Treatment and Rehabilitation Center in Roosevelt, New York. The center was training disabled individuals to use computerized communication aids, specifically adults and children who were unable to speak or had minimal speech due to developmental or acquired impairments. Said Salvatore Gullo, the center's executive director, "With the development of all this new electronic technology, it became apparent that there were more ways to get nonverbal people to communicate and put them in more contact with their environment, with their families and their peers."[1] Another article in the *Wall Street Journal* echoed that hopeful sentiment. It profiled a Long Island man who created a charity to provide pricey communication technologies to nonspeaking autistic children. "It's amazing how difficult life is when you can't communicate," he was quoted as saying, "and this gives them a voice."[2]

The two articles speak volumes about the rhetoric of revolution embraced by technophiles in the digital age,[3] the discourse of technology as an equalizer of access and opportunity for individuals with disabilities,[4] and the notion of "voice" as both symbolizing human speech and serving as a powerful metaphor for agency and self-representation.[5] Each piece sings the praises of consumer electronics companies (Texas Instruments and Apple, respectively) and their mobile communication products with speech output capabilities (the TI Vocaid and the Apple iPad).

However, despite common themes and almost interchangeable quotes, the two pieces were published nearly 30 years apart, in 1983 and 2011. This article explores the linked histories and sociocultural implications of the Vocaid and the iPad. Through this brief case study, I argue that developments in mobile computing and advancements in electronic communication aids for nonspeaking individuals are inherently intertwined through the history of their research, development, commercialization, use, and reuse. Although disability is often underrepresented in the history of computing,[6] it has played, and continues to play, a significant role in how computers augment and provide alternatives to human communication and expression.

Augmenting Mobile Communication History

Many nonspeaking individuals use technologies commonly referred to as augmentative and alternative communication (AAC) devices to augment other forms of communication (such as nonverbal gestures and nonlexical sounds such as laughter) and as an alternative to oral speech.[7] AAC devices range from low-tech (picture cards and plastic communication boards) to high-tech versions (computers like those used most famously by physicist Stephen Hawking and film critic Roger Ebert). Electronic AAC systems provide individuals with significant expressive language impairments (due to disabilities such as autism, cerebral palsy, and traumatic brain injury) with tools for selecting words, symbols, and images to communicate their thoughts and converse with others through digitized and/or synthetic speech.

Prior to microcomputers, electronic AAC devices tended to be stationary and custom built at a cost of $15,000 to $50,000.[8] Early electric communication aids took the form of special systems to control typewriters through alternative inputs (such as a straw that sends signals to a device through inhales and exhales).[8] Priced at $2,000, the Phonic Ear HandiVoice, developed in 1978, was the first portable commercial voice output communication aid.[9] It came in two versions: one with a keyboard for words, pictures, and symbols and another with a calculator-like keyboard that required users to learn hundreds of three-digit codes in order to speak a single word.[10] Contrary to its name, the four-pound HandiVoice was not easily handheld; rather, "saying something with this device was like chiseling words into a stone tablet," noted one user.[11]

Canon and TI were motivated to enter the assistive communication aids market in the late 1970s and early 1980s because of their advancements in microelectronics. US legislation such as the Rehabilitation Act of 1973 and Education for All Handicapped Children Act in 1975, which would purportedly fund such assistive technologies, also incentivized the companies.[12] In 1977, Canon introduced the Canon Communicator, a portable tape typewriter "for non-oral, motor impaired persons."[13] The Communicator was included in a 1980 national print newspaper advertisement commemorating the

continued on p. 93

 Published by the IEEE Computer Society


IEEE computer society
STARTUP
ROCK STARS
Brian Meece
RocketHub
Taryn Rose
DRESR, Inc.
Carey Lai
Intel Capital
Everything You Ever Needed to Know About Startups from the Experts Who Have Made It Happen!
This March 24th event delivers everything entrepreneurs need to propel a successful business, from learning about key legal considerations to bullet-proofing your platform and infrastructure to getting the inside scoop on marketing and funding your startup…
All in One Place for One Action-Packed Day! And You're Invited.
Participate in our Pitchathon—Present your ideas to funders from Intel, HP, Indiegogo, RocketHub, and Crowdfunder, and an audience of potential VC's, Angel Investors and Incubators.
24 MARCH 2015
San Francisco, CA
REGISTER NOW
Special pricing for early registration.
Check it out now before the event sells out! Secure your spot—and your future.
Speakers and judges to include Rock Stars from: Intel, RocketHub, HP, Crowdfunder, Indiegogo, Leonhardt Ventures, DRESR, Dreamitalive.com, Digioh, Honest Dollar and McDermott Will & Emory.
computer.org/Startup
IEEE