

The Origins and Future of Open Source Software

A NetAction White Paper

By Nathan Newman
nathan@netaction.org

NetAction
601 Van Ness Ave., #631
San Francisco, CA 94102

Phone: (415) 775-8674
Fax: (415) 673-3813
Web: <http://www.netaction.org>

This paper is also available as a regular HTML file, a single HTML file, a text file, an RTF file, a postscript file, a PDF file, and this MS Word 5.1 file.

Copyright 1999 by NetAction/The Tides Center. All rights reserved. Material may be reposted or reproduced for non-commercial use provided NetAction is cited as the source. NetAction is a project of The Tides Center, a 501(c)(3) non-profit organization.

In a world where Microsoft increasingly threatens to dominate computing and the Internet, the strongest potential rival to its dominance is no longer its traditional commercial rivals but, surprisingly, a seemingly motley collection of free software tools and operating systems collectively dubbed "open source" software. Unlike most commercial software, the core code of such software can be easily studied by other programmers and improved upon-- the only proviso being that such improvements must also be revealed publicly and distributed freely in a process that encourages continual innovation.

From an operating system called Linux, named for a student from Finland who wrote its core code, to a web server named Apache, put together as literally "a patchy" set of updates to older software by a band of volunteer programmers, these open source programs are emerging not just as inexpensive but as more robust and dynamic alternatives to commercial software.

While this phenomenon surprises some analysts, it should not surprise those with some sense of history. Open source software, largely funded by the federal government, was the wellspring of the creation of the whole computer industry and to this day still lies at the heart of how the Internet came into being. Through a combination of key funding agencies, administrative oversight of software standards and government purchasing rules, the federal government had helped stimulate open source software and open standards for decades. While such software never disappeared, its prominence was undermined by the privatization of the Internet and the commercialization of areas of software once dominated by open source options. Largely, this was due to the fact that in the early 1990s, the federal government pulled back from its commitment to open standards and support for open source software. This left the way open for increases in proprietary, incompatible software and for a company like Microsoft to seek to dominate the computing world with its own proprietary standard.

If open source software is reemerging as an important force, it is largely as a reaction against Microsoft itself. Competitors who themselves have seen their own proprietary alternatives sink under the Microsoft steamroller have suddenly seen alliances with open source software as a chance to halt the Windows monopoly. By itself, this alliance is unlikely to make open source software a real alternative to Microsoft and, more problematically, the opportunism

of the alliance creates a whole set of tensions that need to be resolved for open source software to succeed.

What is needed is a revival of a federal government public policy that supports open source computing and strong standards that can again support the promise of open source innovation. This article will look at the past history of the government's support for open source computing, examine the lessons of its success and the results of its pullback in the early 1990s, and use this history to outline a policy program for the future.

How Government Support Launched the Modern Computing Age¹

One of the earliest threads that led to the Internet and the whole landscape of post-World War II government support for computing began with an Atlantic Monthly article right after the war by Vannevar Bush, a prominent MIT researcher. In that article, he laid out a vision of collaborative science and computing that would spark both economic and technological prominence for America.

Under the psychological impact of Russia's Sputnik success in the 1950s, Vannevar Bush's vision began to take shape as the government sought to regularize its technological research and

¹The history in this chapter derives from a wide range of sources detailed in the bibliography but an invaluable source is: Hafner, Katie and Matthew Lyon. *Where Wizards Stay Up Late: The Origins of the Internet*. Simon & Shuster, New York. 1996 along with Rheingold, Howard. *Tools for Thought--The People and Ideas Behind the Next Computer Revolution*. Simon & Shuster. New York. 1985. Other sources used include: Levy, Steven. *Hackers: Heroes of the Computer Revolution*. Anchor Press. Garden City, New York. 1984; Vinton G. Cerf. "Computer Networking: Global Infrastructure for the 21st Century." Copyright 1995 by Vinton G. Cerf and the Computing Research Association. On the Internet at <http://www.cs.washington.edu/homes/lazowska/cra/networks.html>. Hardy, Henry Edward. *The History of the Net*. Master's Thesis. School of Communications, Grand Valley State University. September 28, 1993; Zakon, Robert Hobbes'. "Hobbes' Internet Timeline v2.5. <http://info.isoc.org/guest/zakon/Internet/History/HIT.html>. 1993-

spending. Given the political biases in the U.S. against government intervention, it seemed inevitable that the engine for industrial policy would be defense-related--even highway and education spending in the period was defined as "defense" to achieve legislative passage.

However, President Eisenhower's personal experience in the military made him distrustful of the bureaucratic interests in the Pentagon, which led him to support the creation of new institutions largely independent of specific military branches. One example was NASA, which ended up with much of the day-to-day applied research of the military, while a new agency called the Advanced Research Projects Agency (ARPA) was created to help coordinate overall R&D spending. Early in the decade, the National Science Foundation was created as a separate agency to fund non-military research, although it would develop a close relationship with the science-based military agencies.

A key appointment at ARPA came in 1962 when psychologist J.C.R. Licklider was hired to head a behavior sciences office, an office that would evolve under Licklider's two-year directorship into the Information Processing Techniques Office (IPTO) which would direct the original creation of the Internet. Licklider in 1960 has written a manifesto for using computers to enhance research and research collaboration called "Man-Computer Symbiosis" and would define the IPTO's office's mandate in research funding. As importantly, Licklider's university background encouraged him and his successors to extend ARPA's funding to a range of university projects.

One key project was a \$3 million per year grant to Project MAC at MIT to encourage the spread of time-sharing computing on the then-breakthrough minicomputer technology. ARPA would fund six of the first twelve time-sharing computer systems in the country, which in turn would help spark the whole minicomputer industry in the 1960s--crucial in the industry and the Boston-area regional economy then but as crucial to the development of the Internet over the next decades. Out of Project MAC would largely develop the early ethos of software and hardware innovation--"hacking" in its early non-pejorative sense before it became confused with electronic vandalism--that launched the computer revolution. It was MIT hackers at Project MAC who largely designed both hardware and software for DEC's breakthrough PDP-6 timesharing minicomputer.

They would spend endless hours creating and sharing new software to extend its capabilities beyond the expectations of its creators.

One of the most radical innovations was the SKETCHPAD program by Ivan Sutherland which allowed the first graphic manipulation of computer images, thereby allowing users to resize and manipulate pictures on the computer screen. Ivan Sutherland would go on to run ARPA and would hire a NASA engineer named Bob Taylor to run the IPTO office after Licklider. Both would use their positions to further promote the creation of breakthrough computing and encourage collaboration across the country.

A key part of this was funding for the Augmentation Research Center (ARC) at the Stanford Research Institute. ARC was run by a researcher Doug Engelbart whose ideas on use of the computer as an aid to individual creativity largely paralleled Licklider's. Taylor pushed through a multi million-dollar grant for computers and staff for ARC's proposed "augmentation laboratory." Out of ARC's lab would come an array of researchers who would go on to become leaders of their own research teams at universities and commercial R&D divisions across the country.

Engelbart worked from Sutherland's precedent to concentrate on using the computer to manipulate text and ideas on the screen. Working with seventeen colleagues and going through three rapid cycles of hardware revolution, by 1968 he was ready to publicly demonstrate the results at an engineering conference called the ACM/IEEE Joint Computer Conference in San Francisco. And the results stunned the audience.

Hooked up by microwave communication to the computers back at SRI, Engelbart would demonstrate the array of tools developed at ARC: the first "mouse" used as an input device, a windowing environment that could rapidly switch between a menu of information sources and models of information, and word processing on screen. None of these had ever been seen before and, in an age when most programmers were still interacting with computers through punch cards, the idea of word processing was a revelation². What was demonstrated was only the showiest example

²Saffo, Paul. "Racing change on a merry-go-round: MIT Management in the Nineties" program reports industry overall is not more productive because of computing technology." Personal Computing

of a set of tools developed to facilitate communication and shared information-based work among intellectual collaborators. ARC was already using text-editing to share common data through hypertext storage (the method of linked pages later used in the World Wide Web) and ran an electronic mail communication system with dedicated e-mail distribution lists among the researchers--all of this years before these innovations would come to the ARPAnet. ARC would also pioneer video-conferencing years before it was developed commercially.

What is startling about Engelbart's achievement, often ignored due to the institutional liquidation of ARC, is how many of the conceptual computing breakthroughs and initial implementations were achieved by his team. To name just a few critical to the networked economy:

- Pioneering distributed electronic mail and e-mail lists five and seven years before ARPAnet.
- Implementing word processing a decade before it began to appear in offices.
- Designing the mouse as an input device sixteen years before Apple introduced it to the world.
- Creating a windowing environment twenty years before Microsoft.
- Envisioning hypertext-linked documents in a distributed environment a quarter-century before the World Wide Web.

All of this was paid for by the federal government due to the vision of Licklider, Sutherland and Bob Taylor at ARPA. As importantly for Silicon Valley, this federal investment would contribute to making the region a magnet for new visionary talent and a wellspring of the networked economy.

How Free, Open Source Software Created the Internet

v14, n5 (May 25, 1990):67. Saffo details the revolutionary vision of Engelbart and how little modern business has engaged with the full thrust of Engelbart's vision.

Out of Project MAC, ARC and other ARPA-funded institutions would emerge the collaborative network that would shape the Internet and computing for the next decades. When ARPA decided to network its various research outlets around the country, it turned to a company called Bolt Beranek and Newman (BBN), a Cambridge-based firm made up largely of MIT graduate students and affiliated researchers (including J.C.R. Licklider at various times). BBN would build the initial network computers needed for what was dubbed the ARPANET while UCLA and ARC would take on administrative duties in managing the network. Four initial "nodes" on the network were linked in October 1969 and by October 1972, when the ARPANET was first demonstrated publicly, there were twenty-nine nodes in the network. What would evolve into the Internet had been born.

ARPA would oversee the creation of an array of software needed to manage and extend the computer network. The first set of standards, now known as the "Request for Comments," or RFCs, was the work of the late Jonathan B. Postel. The first standard network protocol was created in 1971 to allow a person at one computer to connect to other computers on the network as if they were local users. This soon evolved into the standard Transmission Control Protocol (TCP) which was complemented in 1972 by the File Transfer Protocol which allowed individual files to be exchanged between computers. In 1976, ARPA hired Vint Cerf, a Stanford professor and an original member of the UCLA graduate student group which helped launch the ARPANET, and Bob Kahn, a former BBN manager on the project, to create a system for integrating the ARPANET with other computer networks. By 1977, they had demonstrated the Internet Protocol (IP) which could be used to integrate satellite, packet radio and the ARPANET. From this point on, new networks of computers could be easily added to the network. In 1981, ARPA funded researchers at UC-Berkeley to include TCP/IP networking protocols into UCB's popular version of the UNIX operating system, thereby spreading the Internet standards to computers throughout the world.

An example of the cross-fertilization of staff and ideas outside the government was the case of Bob Metcalfe and Ethernet. Bob Metcalfe had originally designed the interface to connect MIT's computers to the ARPANET and had been hired in the mid-1970s at Xerox Corporation's new Palo Alto Research Center (PARC) which was headed by Bob Taylor, the former IPTO head who had started the ARPANET project. Metcalfe was doing ARPA-funded work while

trying to figure out how to cheaply network PARC's experimental personal computers. Using models from ARPA's project around radio packet switching, Metcalfe created a system called Ethernet to exchange information between computers in what would come to be called Local Area Networks (LANs). Ethernet was crucial for the expansion of the Internet since local computers could be networked together and then connected to other networks using the TCP/IP protocol and local router computers. Xerox would start selling Ethernet as a commercial product in 1980 (and Metcalfe would found 3Com to sell networking technology), while PARC head Bob Taylor donated millions of dollars of Ethernet equipment to universities to help expand use of networking on campuses.

In all these ways, ARPA helped shepherd open Internet standards into the 1980s and 1990s when they would be used to radically expand the network to a wide range of users. In doing so, it was clear that the professional norms promoted by ARPA and the community of researchers was critical in order to keep individual profit-taking from undermining those open standards. As one example, in 1973 then IPTO head Larry Roberts was hired by BBN to run a company subsidiary called TELENET that would run private packet switching networks. In coming to BBN, Roberts carefully deflected a bid by BBN to take over ARPANET privately. J.C.R. Licklider, who returned from MIT to ARPA to replace Roberts as head of IPTO, soon found himself in conflict with his old employer, BBN, which was refusing to publish the original computer code for the IMP computer routers which the company had designed. Making matters worse, BBN was becoming more and more reluctant itself to fix software bugs faced by the system (no doubt preferring to concentrate on programming for its for-profit TELENET subsidiary). Licklider, in the name of the openness of the Net, threatened to hold up BBN's federal contract funds unless the company released the code publicly. BBN did so, thereby enhancing--albeit reluctantly--the tradition of open codes in the development of standards.

A key part of the success of the Internet was the fact that the public space of the network harnessed the energy of universities, both paid staff and volunteers, to provide a continuous stream of open source software to improve its functionality. The Net itself allowed any new innovation to nearly instantaneously ricochet across the nation, even the world, without the friction of the costs of either distribution or purchase. This "gift" economy allowed new innovations to be quickly tested and to gain a critical mass of users

for functions which had not even been envisioned by the creators of the system.

The ethic of shared, open software, what was called the "hacker ethic³," at MIT's Project Mac, would contribute to both the creation of the Internet and the spread of computing across the country. Many early efforts were games like Spacewar and Adventure, but more serious software became the staples of day-to-day computing throughout the Internet and beyond. Probably the most pervasive example was the early use of the ARPANET for electronic mail. Not even planned as part of its design, email was created as a private "hack" by BBN engineer Ray Tomlinson in 1972 as a piggyback on the file transfer protocol. Under the tolerant supervision of ARPA, use of the Net for email communication soon surpassed computing resource sharing. Stephen Lukasik, ARPA director from 1971 to 1975, saw the importance of email for long-distance collaboration and himself soon began virtually directing ARPA from electronic mail and his 20-lb Texas Instruments portable terminal. Partly because of Lukasik's own frustration in dealing with the stream of raw mail, IPTO director Larry Roberts himself wrote the code for the first mail manager software, called READ. This was soon supplanted by the popular MSG, which added the first reply function. New free email managers have been a staple of Internet innovation ever since. Eric Allman, a student at UC-Berkeley, would create the program SENDMAIL to assist network managers in directing and processing this ever increasing email traffic--to this day, Allman's program is used to direct over 75% of Internet email traffic. Others at Berkeley created the Berkeley Internet Name Daemon (BIND) program which is used to direct traffic to sites by the site name--like www.netaction.org--rather than having to use numbers like 199.201.243.200.

While free and open source software continued to enhance the spread of computer networking, what ultimately brought the Internet into its own was the "Gopher" software developed at the University of Minnesota in the early 1990s. Building on the existence

³The term "hacker" was originally a descriptive term which implied a shared belief that technical information should, in principle, be freely available to all users. While some individuals who promoted this perspective are still actively involved in Internet development, the term has a different meaning to the current generation of Internet users.

of individual Internet sites where files and programs could be retrieved after logging into a particular computer over the network, Gopher was a piece of software that could be used to create personalized lists of files from computers all over the Net and allow computer users to view or retrieve any file chosen from the list. With this innovation, the Internet became one giant hard drive that could be organized and presented to a particular set of users in whatever way made the most logical or aesthetic sense. Gophers sprang up on computers run by governments, universities, community organizations and businesses which were beginning to stake a place on the Net. In a visual way, the Internet's vast resources could be presented and reached through Minnesota's "All the Gopher Sites in the World" gopher site. For most commercial users of service providers like America Online, gophers were the user's initial contact with the world of the Internet, and this contact created demand for even more of the content that users knew existed out of the proprietary walls of those commercial providers.

The next step, and the step that brought the Internet into almost daily headlines, was the World Wide Web. The Web was initially designed at the European Particle Physics Lab (CERN) in Geneva, Switzerland to share information internally--what would be designated as an Intranet today. However, people quickly saw it as a useful way of sharing information between computer systems much like the Gopher software, with the additional advantage of "hypertext" connections to internal parts of documents. In 1993, computer science students funded at the National Center for Supercomputing Applications located at the University of Illinois created Mosaic, the first Web browser that added the display of graphics to the traditional text display. With an almost unnerving speed, Web sites exploded across the Internet along with the browsers needed to view them. It was only with Netscape's creation of its Navigator software, followed soon by Microsoft's Explorer software, that secret code and commercial software began to erode the open source tradition of the Internet--an issue we will return to later in this paper.

Supervision and Standards

If government funding helped support new software as a font of innovation on the Internet, government supervision helped maintain the standardization required for easy compatibility between the wide range of computers increasingly sharing resources on the Net.

Despite odes to the "anarchy" of the Internet, its creation was a closely supervised anarchy directed to the specifications of government, yet marshaling the broad professional, volunteer and eventually commercial resources of the emerging computer elite. In many ways, the very skill of the government in marshaling those resources with a light hand is a source of the sometimes rhetorical amnesia over its role. The smoothness of the Internet's creation and the building of a broad consensus over its shape created so much legitimacy for its design that it was seen less as a creation of "the government"--i.e. "them"--and more as a creation of society as a whole.

Licklider had actually started this professional network at ARPA in the early 1960s when he reached beyond traditional experts at federal agencies and national labs to gather an association of experts interested in communication technology. He oriented ARPA to establish contacts with university researchers around the country, establishing what he presciently called the Intergalactic Computer Network which helped connect researchers interested in computer networking.

When ARPANET was created, UCLA was funded to establish a Network Measurement Center to oversee the evolution of the network. Forty grad students at UCLA, many of them to become key leaders in both the public and corporate Internet world, helped run the center and coordinate with other researchers around developing the standards for running the ARPANET. The new technology itself helped add a whole nationwide group of researchers and graduate students in these deliberations to help mold the evolution of the Internet. This national body became the Network Working Group (NWG) which was expanded after the 1972 "debut" conference to become part of an International Network Working Group to promote international computer networking.

Management of Internet "addresses," critical for the decentralized electronic switching network, would be housed at Doug Engelbart's shop at the Stanford Research Institute in an institution called the InterNIC. As the NIC, Engelbart would help identify and organize electronic resources on the Internet for the easiest retrieval. Until 1992 (when the NIC functions were awarded to other companies), the function of the NIC at SRI would include administration in assigning IP network addresses and domain names for all servers,

essentially creating the yellow pages for the Internet.⁴ Surveying the initial implementation of the ARPAnet in a speech in 1970, Engelbart could already envision the evolution of the networked community where, "there will emerge a new 'marketplace,' representing fantastic wealth in commodities of knowledge, service, information, processing, storage, etc."⁵

ARPA would replace the NWG by a more formal Internet Configuration Control Board (ICCB) in 1979 to extend participation in the design of the Internet to a wider range of members of the research community. This was especially important as the ARPANET expanded to include a range of other government agencies and bodies and evolved into the diversity of the emerging Internet community. The ICCB was later replaced by the Internet Activities Board (IAB) which used a set of ten task forces to include a wide range of experts in the evolution of the Internet. As the Internet was privatized in the early 1990s, the private sector (led in many cases by former researchers for ARPA and its Internet-related funded projects) created the Internet Society in 1992, and the IAB reconstituted itself as the Internet Architecture Board and joined the Internet Society.⁶

At each step of its development, ARPA and associated government agencies expanded participation to an ever widening set of experts and technological leaders who, in turn, would encourage others in their academic, scientific, community or business realm to support the effective development of the Internet. As well, the continual movement of personnel back and forth from academic, government and (eventually) business positions created a cross-fertilization of ideas and a loyalty to the emerging network rather than to any particular organization.

It was the weakening of this government-supervised network of standards in the 1990s that allowed commercial competition over standards to undermine open computing, setting the stage for both

⁴Baker, Steven. "The evolving Internet backbone--history of the Internet computer network." UNIX Review v11, n9 (Sept, 1993):15.

⁵Rheingold, 1985, p. 199.

⁶There is a more extensive history of this evolution of professional governance of the Internet in: Kahn, Robert E. "The role of government in the evolution of the Internet." Communications of the ACM v37, n8 (Aug 1994):15-19.

the Netscape-Microsoft browser war and for Microsoft's overall expanding monopoly on standards (issues we will return to later).

UNIX as a Public Standard

If anything illustrates both the gains from government support of open standards in computing and the dangers from public policy withdrawing from that support, it is the UNIX operating system.

UNIX was the first operating system developed that was independent of specific hardware, thereby giving users and programmers freedom from the dictates of hardware designers. UNIX could be "ported" to different machines, thereby allowing the same program to run on completely different hardware. Created at Bell Labs in the late 60s when AT&T was still barred from the computer business, UNIX was widely licensed by AT&T, mostly to universities. UNIX was especially popular with ARPANET programmers working on a wide variety of computers because they needed to create an integrated set of software tools for managing their emerging network.

UNIX had developed during the 1970s into a number of lackluster variations, so in the late 1970s, UC-Berkeley researchers--funded largely by ARPA--developed an improved version that was dubbed UNIX 4.1 BSD (Berkeley Software Distribution). Bill Joy, the lead programmer in the Berkeley UNIX effort, was again funded by ARPA in 1981 to create a new version of UNIX including TCP/IP networking protocols. With a minimal licensing fee, Berkeley seeded its UNIX version with its Internet protocols throughout the university world.

Probably no single private company benefitted more from (and contributed more to) the open UNIX and Internet standards than Sun Microsystems, a seller in the early 1980s of new high performance machines dubbed workstations. Sun would enter, then dominate, the market for stand-alone workstations that were beginning to replace time-share minicomputers. Started in 1982, Sun would be one of the fastest growing companies in history, making the Fortune 500 within five years. By 1995, the company would sell 1.5 million high performance computers, used as the core systems for networking in government, universities, finance and engineering. And from the first day of operation, every Sun computer was shipped with UNIX with hardware and software designed to be hooked up to the Internet. It was on Sun UNIX machines that much of the Internet would be networked in the

1980s, and it was on Sun Workstations that the first Web browser, Mosaic, would be designed.

That Sun was committed to open standards reflected the company founders' emergence out of the milieu of Bay Area graduate students immersed in the ARPANET. When Stanford M.B.A.'s Scott McNealy and Vinod Khlosa teamed up with Stanford student Andy Bechtolsheim, who had developed a new high performance computer using off-the-shelf components, it was natural for them to adopt UNIX, the popular university operating system, as the operating system for their new computer. And it was natural for them to bring in as a co-founder Bill Joy, the premiere UNIX and ARPAnet programmer at UC-Berkeley.

Commercial versions of UNIX, however, were splintered between various incompatible proprietary versions. Far from being a widely used standard in business that Sun could just hop a ride on, Bill Joy and the Sun team had to help build a standard and sell private industry on the gospel of open computing. They took a number of steps to ensure that the BSD UNIX on Sun's computers was seen as a real standard. Sun gave away the BSD UNIX and TCP/IP networking software with every computer they sold. When Sun developed a Network File System (NFS) in 1984 that enhanced network computing by making it possible to share files between different computers, they didn't try to sell this advance as normal software. Instead, they licensed it to the industry for a nominal licensing fee and even published the specifications for the software on the usenet electronic bulletin board so anyone could construct an alternative to the NFS file system if they wanted to avoid the license fee. Usable on DOS, VMS and other operating systems, it was a key advance for networking and increased trust by customers that Sun would be an honest guardian of the open standards it was promoting on its hardware. Another key step was made in 1985 when Sun approached AT&T, allowed back in the computer industry, and worked out an agreement to merge Sun's Berkeley UNIX with AT&T's System V, further enhancing the public view of Sun's UNIX as the standard.

The key for making UNIX nearly universal in corporate and high-end computing in the late 1980s, though, was decisive action by the federal government in support of strong UNIX standards. The federal government itself was faced with a mess of different computer systems that needed to be networked together. Because of the close

ties of the Department of Defense to university researchers (largely fostered by ARPA/DARPA), the federal government already had an affinity for UNIX. So in 1986, the government passed regulations that no company could bid on any government computer contract unless their system offered UNIX as an option. This gave Sun a huge advantage in securing a large slice of the \$500 million, five-year National Security Agency contract then under bid. Sun's and AT&T's version of UNIX was now the benchmark for selling to the government and university markets (along with many private industry customers who would follow the government's lead in standards). This was reinforced in 1988 when the Air Force declared DEC's proprietary version of UNIX, called Ultrix, ineligible for government contracts.

Other workstation and corporate computer makers would do a complete turnaround in 1987 and 1988 and begin promoting their own "open computing" UNIX systems--all with the built-in Internet protocols that would set the stage for the commercial explosion of the Internet in the 1990s.

Unfortunately, this was also the period of government withdrawal from strong support for computing standards and the result was the development of different UNIX standards, as Sun, Hewlett Packard and other companies lined up behind different variants in commercial warfare. This fragmentation of UNIX standards was soon mirrored in the war between Netscape and Microsoft over Internet standards that followed the government's withdrawal from defense of those standards.

Breakdown of Open Computing on the Internet

It was with the World Wide Web that the Internet broke into national consciousness and where Netscape Communications would become the central Bay Area firm around which a slew of new Silicon Valley companies would form. But unlike Sun, which rode public UNIX standards to rapid growth, Netscape began its life with a direct assault on the original government-based standards created by the National Center for Supercomputing Applications (NCSA). In this, Netscape would play a three-cornered game against both the NCSA and Microsoft, who it knew would quickly be coming in with its own controlled standards. Netscape's success would be based on the virtual withdrawal of the government from any serious intervention on behalf of Internet standards.

The initial Web "browser," Mosaic, was created at the University of Illinois at Champaign-Urbana where the National Center for Supercomputing Applications (NCSA) was located. The National Science Foundation had officially funded the NSFnet "backbone" of the Internet to link five major supercomputing centers, including NCSA, and NCSA's software development group had concentrated for years on high-performance information-sharing and collaboration software. Even before Mosaic, the NCSA had back in 1985 created software "clients" for PCs and Macs, called Telnet, to allow people to access and use computers connected to the Internet as if the user were locally based. A different computer center at Illinois was responsible, as well, for the popular Eudora client for electronic mail on PCs and Macs. The NCSA had worked to create a graphics-based collaborative tool for sharing documents called Collage, so it was natural for them to create a team to develop a graphics-based version of the Web "HyperText Markup Language" (HTML) protocols created by CERN in Europe.⁷ The result of this forty-member team was Mosaic, first introduced on the UNIX platform in January 1993, with Macintosh and PC versions introduced in August 1993. Copyrighted by the University of Illinois, Mosaic could be downloaded for free by individuals and by companies wishing to use the Internet for internal communications.

However, the NCSA did not want to become a help desk for commercial applications, so in August 1994, the University of Illinois assigned future commercial rights for licensing NCSA Mosaic to Spyglass, Inc., a local company created by NCSA alumni to commercialize NCSA technology. The goal was for university researchers to continue developing longer-term technology and standards to be incorporated into browsers, while Spyglass would help license the technology to companies addressing immediate customer needs such as support, speed, and security. Spyglass began widely licensing Mosaic to computer companies including IBM, DEC, AT&T, NEC, and Firefox Inc., who was working to integrate Mosaic standards into Novell networking software for the personal computer.⁸

⁷Michalski, Jerry. "O pioneers!". RELease 1.0 v94, n1 (Jan 31, 1994):5 (8 pages).

⁸Stevens, Tim. "NCSA: National Center for Supercomputing Applications." Industry Week v243, n23 (Dec 19, 1994):56-58 and

Watching Mosaic from the Bay Area, Silicon Graphics CEO Jim Clark, a veteran of the UNIX standards wars, understood how much money could be won if a company could take control of the standards of this new Internet tool. So Clark left his company and set out to destroy Mosaic and replace its government-backed standards. He met with Marc Andreessen, a member of the Mosaic team who had been hired at a Bay Area Internet security firm called Enterprise Integration Technologies. Out of that meeting in April 1994 was born Mosaic Communications Corporation (later to be called Netscape). With Clark putting up the capital, Andreessen recruited five other Mosaic team members from NCSA to design what they called in-house Mozilla, the Mosaic-Killer. In six months, Clark's team had created a powerful browser, which the team called Netscape. It had easy-to-navigate features and loaded graphic images faster than NCSA's Mosaic. But Netscape did something else--it included the ability to display text formatting that did not even exist in the HTML standards embedded in the NCSA Mosaic browser. This meant that Web pages designed to work with Netscape would not be readable by all the other Mosaic-based browsers. This would encourage people to use Netscape browsers and, as Netscape developed them, would encourage Web designers to pay Netscape for the server software that developed Web pages using their modified standards. It was in this later market of selling Web design tools costing from \$1,500 to \$50,000 where Netscape intended to make their money.⁹

And then Clark and Andreessen compounded their fracturing of the NCSA standard by giving their version away over the Internet. The University of Illinois had demanded that Clark's company pay for a license before selling their version. Clark later said that he refused because the university was demanding an ongoing per-copy royalty: "I didn't tell them, but we had intended to allow people to download it, and they were going to charge me. The amount varied, but nothing is innocuous when you're talking tens of millions of

Patch, Kimberly. "Spyglass takes on Mosaic licensing: will focus on support and security." PC Week v11, n34 (August 29, 1994):123.

⁹Accounts of Netscape's startup from: Holzinger, Albert G.

"Netscape founder points, and it clicks." Nation's Business v84, n1 (Jan 1996):32; Nee, Eric."Jim Clark." Upside v7, n10 (Oct 1995):28-48.

people."¹⁰ The point of the licenses by Illinois had been, along with collecting a little revenue, to control the standards and make sure that the only free version available was the official NCSA standard. Netscape would essentially "dump" its version onto the Internet, thereby undercutting the rest of the commercial browser companies, which couldn't duplicate Netscape's actions because they were fairly paying per copy license fees. So Netscape, being the sole enhanced commercial browser flooding the Internet, was able to destroy NCSA-led standards and take over standards creation itself.

Unlike the situation with Sun Microsystems, where the government would decisively support open government-based UNIX standards, the federal government did nothing to support NCSA's standards. Other companies and analysts would immediately condemn Netscape's actions as a monopolistic move¹¹, but the government made no investigations into possible monopoly practices, no lawsuit alleging intellectual property infringement, no announcements that the federal government would use only NCSA-approved codes in government Web sites, no announcements that it would refuse to buy any Web servers (i.e. Netscape's) based on such non-standard formatting, and no signal from the government at all that they would oppose Netscape's takeover of the standards. Instead, the University of Illinois, after a bit of public grumbling, threw in the towel. They signed an agreement with Clark in December 1994 that allowed Netscape to be sold without a license for the minor concessions that the words "Mosaic" be removed from the firm's title and that no mention of Mosaic be made in marketing the browser.¹²

In a perverse way, Clark and Netscape would justify their destruction of the government standards based on the expected weakness of the government in defending them. They predicted that Microsoft would soon use its dissemination of the operating system to take control of

¹⁰Nee, Oct 1995.

¹¹Steinert-Threlkeld, Tom. "The Internet shouldn't be a breeding ground for monopolies--Mosaic Communications' NetScape giveaway could be prelude to market dominance." *InterActive Week* v1, n2 (Nov 7, 1994):44.

¹²University of Illinois and Netscape Communications reach agreement." *Information Today* v12, n3 (Mar 1995):39.

standards if Netscape didn't do so first through free distribution. Argued Clark:

At some level, standards certainly play a role, but the real issue is that there is a set of people, a set of very powerful companies out there, who don't play the standards game. For the standards game to work, everyone has to play it, everyone has to acknowledge it's the game. Companies such as Microsoft aren't going to sit around and wait for some standards body to tell them. If your philosophy is to adhere to the standards, the guy who just does the de facto thing that serves the market need instantly has got an advantage.¹³

Netscape, having seized leadership of Web standards, would try to redeem its reputation by working with the old Internet fellowship of engineers embodied in the Internet Engineering Task Force (IETF) and the more recent World-Wide Web Consortium (W3C) based at MIT and run by CERN's Tim Berners-Lee, who came to MIT in late 1994.

And as Microsoft entered the game with its own Internet Explorer browser to appear on every Windows desktop, the grumblings over Netscape's occasional forays into proprietary advantage would lessen as the alternative fear of Microsoft taking over the whole computing world loomed. Having come late to the Internet, Microsoft initially directly licensed Mosaic browser technology from Spyglass in December 1994--a license netting Spyglass about \$13.1 million. But when Microsoft began giving its browser away at the end of 1995, the rest of Spyglass's licensing revenue (amounting to \$20 million) disappeared as the browser war settled into a two-company fight between Netscape and Microsoft.¹⁴

In the end, Netscape would argue that the beloved public village of standards was threatened by Microsoft, and Netscape had only destroyed the village in order to try to save it. With the government withdrawing from its role in defending standards, such a standards war was inevitable.

¹³Nee, *ibid.* 1995

¹⁴Lohr, Steve. "Spyglass, a Pioneer, Learns Hard Lessons About Microsoft" *New York Times*. March 2. 1998.

Why the Government Withdrew from Defense of Open Standards

So why the withdrawal by the government in the first place? The retreat of federal involvement has been based on a combination of ideological opposition, private industry desires, and the disappearance of a stable government bureaucracy able to assume the role of regulator. This has left Internet development increasingly in the hands of self-interested companies seeking commercial advantage rather than maximum innovation and compatibility for consumers.

The ideological assault on federal involvement in further developments of the Internet is strongly related to the end of the Cold War and the withdrawal of the "national security" basis for much of the federal government's economic involvement since World War II. It was probably not a coincidence that ARPA director Craig Fields, criticized for ARPA's involvement in trying to direct the development of high technology, was fired by the Bush Administration in 1989--the same year as the fall of the Berlin Wall. While the Clinton administration made some gestures in asserting a public interest in the development of what they called the National Information Infrastructure, privatization proceeded apace. What limited funds the Clinton Administration allocated for encouraging community and local government development of the Internet was vociferously opposed by conservatives in Congress and, with the Republican takeover of the Congress in 1994, those funds were initially zeroed out and in the end sharply limited, even as local need for the funds exploded with the expansion of the Net.

As for Internet standards, criticism had already been leveled against the University of Illinois and NCSA for attempting to manage the expansion of the World Wide Web¹⁵ and, in the context of Newt Gingrich's anti- government message, there was probably even less support for government regulation of standards.

Private industry had significantly benefited from government spending on the Internet in the period when it was not commercially viable and the government was the main market for Internet-related computer services. However, as a private market for Internet

¹⁵Messmer, Ellen. "Spyglass captures Mosaic licensing." Network World v11, n35 (Aug 29, 1994):4.

services appeared around the structure of the Internet, private industry has seen strong government involvement as a threat to corporate control of information markets. Companies that had started life as extensions of the government saw the opportunity for independence and extremely high profits as the government's role receded. The success of government intervention in nurturing new economic sectors is often rewarded by the creation of a private sector interest in blocking further government action.

Similarly, the success of the private sector helped fragment and undermine the ability of key government agencies to successfully promote the public's interest. Partly, this is due to ideological opposition from business, which politically sought to curtail the power of the public sector as the private sector expanded commercially. With Defense involvement in high technology under assault, and Republicans trying to abolish the Commerce Department where most of the NII programs have been coordinated in the Clinton administration, there has been little chance to consider the long-term potential for public servants watching their political backs.¹⁶ Also significant was the movement of ARPA employees from public service to private companies now pushing for limiting the federal role. From Bob Metcalfe, who became rich through founding 3Com, to Vint Cerf, who has become a major spokesperson for MCI, the founders of ARPANET who initially cultivated the ethic of freely sharing information and software are now fighting for profit share and private ownership of intellectual property.

The Return of Open Source Computing

So in the midst of UNIX wars, browser wars and commercial competition, the emergence of open source software as a more and more accepted part of the computing environment comes as something of a surprise. The catalyst is Microsoft, or rather the reaction of Microsoft's Silicon Valley competitors to the Seattle-based company's monopolistic practices. The problem for Silicon Valley firms, despite the pride in the geography-driven technology innovation of the region, is that such proximity does not

¹⁶See Hellerstein, Judith. "The NTIA needs to rethink its role in the new telecommunications environment." *Telecommunications (Americas Edition)* v30, n8 (Aug 1996):22 for the trials of the NTIA agency in the Clinton administration.

automatically create the standards that propel economic growth, especially in the absence of a firm alliance with government. Technology firms have tried to create substitutes for government through private consortia like CommerceNet and other standards bodies, but none have the core of public-interested officials that government can wield to transcend particularistic company concerns in favor of the public interest.

The reality is that despite the Internet's success, many of the firms in the region continually struggle with the danger of proprietary technologies upsetting the trust needed to sustain the collaborative model that has fueled the growth of firms in the region. At the top of the list of dangers is of, course, Microsoft. Microsoft used a combination of its early alliance with IBM and hardball tactics to build its proprietary operating system monopoly on the desktop. From that base, Microsoft would extend its proprietary standards into the market for large-scale business computing, formerly the province of mainframes or UNIX-based network servers. While the Internet at first appeared as a danger to Microsoft, even a dagger at its throat, Microsoft also saw that success in molding those standards in a proprietary direction could extend the company's control throughout the whole world of corporate computing.

Microsoft responded with a combination of in-house software applications and developer tools optimized for its proprietary standards, creating an all-pervasive computing environment that promised any corporation that its needs would be met. The Microsoft solution might be less innovative than any particular competitor, but Microsoft's very completeness and pervasiveness across all sectors of computing would make up for its rigidity.

In fact, Microsoft's rigidity could be an advantage when compared to the weak standards that pervaded the UNIX corporate environment by the 1990s. After the heyday of the 80s when government purchasing requirements had enforced a broad UNIX standard on the industry, the industry had divided into warring UNIX camps and left customers uncertain that their needs would be met in the fragmented UNIX environment. By 1997 Microsoft NT computer servers were outselling UNIX servers.¹⁷ It was clear that in the absence of strong standards and government support for such

¹⁷McGarvey, Joe. "Intranets, NT Shape Server Market." SoftBase. Jun 1, 1997.

standards, proprietary models had a decided advantage in yielding the market stability and monopoly rents that a company like Microsoft could reap.

Even as Silicon Valley firms sought to finesse innovation from the economic pressure of Windows competition, the UNIX wars made clear that strong open standards were the key to the region competing economically against proprietary steamrollers like Microsoft. Consortia like CommerceNet built around Internet standards were the first step in the process, but companies like Sun and Netscape saw the need for broader solutions that would expand open standards from the operating system to the tools used by programmers. The Java language, with its promise that any program would be able to run on any computer, no matter its hardware or operating system, became a part of that strategy.

With the need to generate stronger global support for its standards, Netscape took the unprecedented step in March 1998 of publicly revealing its browser source code--the usually top-secret guts of any program. Netscape invited developers to modify the code and even resell their own version as long as any modifications to the code were republished publicly and made according to the terms of their license, and subject to coordination by the development team at Mozilla.org. Faced with the onslaught of Microsoft's proprietary approach, Netscape decided that the regional commercial commitment to developing standards was insufficient. It needed to marshal the resources of the global programming community, and it needed to open its code to gain the kind of trust needed to ensure their support.

The idea, harking back to the original ARPANET vision, was to invite the participation of the whole Net community in developing the tools and standards embedded in the browser software. "It's no longer Netscape alone, pushing the client software forward, but now it's really the whole Net," said Bob Lisbonne, Netscape's senior vice president for client products at the time. "For Netscape, this gives us a way to engage the creative, innovative abilities of literally orders of magnitude more people than we could ever--really any commercial software company could ever afford to just put on their payroll.¹⁸" With hacker enthusiasts lauding the decision, thousands

¹⁸Kornblum, Janet. "Netscape sets source code free." News.Com. March 31, 1998.

of developers would download the source code within the first day and major modifications of Navigator were released onto the Net within weeks by independent developers from all over the country. The idea was that Netscape could release existing modifications in its continual upgrades of both browser and server software. What it lost by giving up control of its code, it would make up through selling customized business versions and server software, and by preventing Microsoft's control of standards which would be Netscape's deathnell.

Netscape's action highlights the continued importance of public-interest-oriented software development. This type of software has survived much of the privatization of the Internet. Most dramatically, despite the focus on the Microsoft-Netscape rivalry, the most popular Web server on the Internet was neither company's but rather a free, open source server called Apache. After the NCSA developed its Mosaic browser software and its original server software, the NCSA as part of the government privatization had ceased aggressively updating its software. Instead, a geographically dispersed group of software programmers, some at universities and some in private business, began collaborating in 1995 to update the NCSA server to increase its power and manageability. Most of the programmers participated out of altruism. The result was a Web server that in 1997 was used in 44 percent of Internet sites, compared to just 16 percent that used Microsoft and 12 percent using Netscape's software. And that list of sites included McDonalds, UUNET Technologies, HotWired, Yahoo Inc., CSB, the FBI and IBM, which passed over its own Lotus Domino server in favor of Apache when it put its "Big Blue-Gary Kasparov" chess match on the Internet.¹⁹ Similarly, one of the favorite Web programming languages is a free and open language named perl which has similarly been modified and improved through a global network of collaborators coordinated by programmer Larry Wall.

Netscape also announced that it would begin making all its software applications available in the Linux operating system, a freeware version of UNIX that has become the fastest expanding operating system in the world with three to nine million copies on computers around the world. Linux was described by Wired magazine in 1997

¹⁹Moeller, Michael. "Fort Apache: freeware's spirit outshines commercial products." PC Week v14, n23. June 9, 1997.

as "[Window] NT's most serious competitor, the only viable alternative to the Microsoft monoculture."²⁰

Remarkably, Linux was born in 1991 by a student at the University of Helsinki in Finland whose first name, Linus, led to the naming of the language. At that point, a whole series of free and open source UNIX tools had been developed by programmers connected to GNU (a self-referencing title standing for GNU is Not Unix) foundation, itself founded by one of the original MIT hackers, Richard Stallman, who objected to the increasing commercialization of university research. Stallman and his fellow GNU hackers had, rightly, feared that despite the fact that popular UNIX standards like Sun's were broadly distributed, they still remained under private ownership and could and would be used for proprietary advantage under the right (or wrong) circumstances. Which is what happened by the early 90s.

The community of GNU programmers and users sought a non-proprietary UNIX alternative to escape the new UNIX standards wars between competing commercial providers. What this network of free software developers lacked was the core of the operating system, called a "kernel," which would tie all the GNU UNIX tools together into an alternative to the commercial UNIX competitors. Linus Torvalds wrote that kernel and from his university post would use the Internet to coordinate improvements in this new operating system with help from hundreds of enthusiasts around the globe.

Based on what GNU called "copyleft" principles, the Linux operating system could be distributed freely or packaged with documentation and sold for modest amounts backed by technical support by companies like Red Hat, Caldera and Cygnus Solutions. Extremely popular in developing nations like South Africa, Cuba, India and the Philippines, Linux also began to eclipse other forms of UNIX in the U.S. partly because of its price but also because many people considered it technically the best operating system in existence. Linux was the first operating system to include Java capability, so every increase in Java programs adds to its functionality.²¹

²⁰Glyn Moody. "The Greatest OS That (N)ever Was." Wired. August 1997.

²¹Sullivan, Eamonn "Freedom is priceless, even when it's free." PC Week v13, n47. Nov 25, 1996.

Netscape's source code unveiling, and its announced support for Linux, throws into relief the different economies of trust that separate proprietary standards and open source standards. With proprietary standards like Microsoft's, everyone trusts (or fears) that Microsoft will enforce whatever standards it dictates from its company-specific development. The result is that hardware and software partners on such proprietary efforts develop products anywhere to their uniform standard. Alternatively, collaborators on open source software can increasingly use the Internet to build trust based on altruism and the hacker ethic of achievement without needing to share any geography--the extreme example being Linus Torvalds' direction of the evolution of Linux. Without expectation of financially capturing the social benefits of their creation, they are free to innovate without restriction. On the other hand, with more diffuse commercial standards, collaborators need the repeated interactions and day-to-day commercial interactions of shared geography, like Silicon Valley, to generate financial gain while assuring that multiple collaborators all profit from innovation.

With Microsoft's proprietary approaches gaining ground, Netscape and other Silicon Valley actors reluctantly saw their alternative commercial standards losing ground and saw an alliance with the global open source software model as necessary for survival. They would forgo some profits in order to maintain the priority on innovation that gives them an advantage in the remaining commercial aspects of technology development.

A range of new partners to Linux and other open source software have emerged. Corel--maker of the WordPerfect wordprocessor--announced it would be releasing a full suite of office applications for Linux. Inprise (formerly Borland) announced that its Interbase database server would be ported to Linux. IBM announced that its next set of Web tools, called WebSphere Application Server, would fully support the Apache web server. IBM also announced it would join the Apache Group of collaborative developers and be contributing code to improve the Apache server. Sendmail creator Eric Allman has created a startup business to sell easy-to-use administrative tools to support the core free Sendmail program. Hewlett-Packard, Compaq, IBM, and Silicon Graphics have all indicated plans to install and support Linux for their hardware customers. Lotus will release a version of its Domino and Notes collaborative software for Linux later in 1999.

Where to Go from Here--Ending the Microsoft Monopoly

So with all this good news, open source software might seem to be an antidote to the threat of Microsoft's monopoly. At least that was the argument Microsoft executives were making in court in January. Of course, even Microsoft's in-house political magazine, Slate, noted the irony that everything the executives described about "Windows' impending obsolescence and its rivals' virtues [was] exactly the opposite of what Microsoft tells consumers and corporate clients."²²

Microsoft may play up the marginal danger of Linux to its market share, but the sobering reality is that Microsoft server sales are still growing faster than the overall market growth-- increasing Microsoft's market share. Linux is growing, but mostly at the expense of commercial versions of UNIX, whose growth has essentially dropped to a standstill of only 4% in 1998. And neither Linux nor a rejuvenated Apple is undermining Microsoft's complete domination of the desktop market.

While many of Microsoft's competitors are collaborating in support of Linux, they are just as likely--especially with a little Microsoft incentive--to fall out into competing camps that might easily divert Linux or other open source software into a competing muddle of standards. Just as the UNIX wars served Microsoft, a similar split in the future could easily knock Linux out as a strong competitor to the centrally-controlled standards of Windows.

However, the limited success of Linux and other open source software does have implications for the Microsoft antitrust trial. The availability of open source software is not an excuse to find Microsoft innocent of the wholesale monopolistic abuses that the trial has exposed. But it may become one of the remedies that the court and other government agencies use to rein in Microsoft's monopoly power.

Many, including NetAction, have proposed remedies to Microsoft's monopoly abuses, from breaking up the company into multiple competing units to court-ordered limits on its licensing agreements to forcing Microsoft to reveal its source code to prevent in-house coders from having advantages over competitors in non-OS markets.

²²Saletan, William "Microsoft plays dead." Slate January 29, 1999.

While such government restrictions are likely necessary, none of them speak to the issue of creating a strong viable alternative to Microsoft. As Mitch Stoltz notes in "The Case for Government Promotion of Open Source Software," the federal government already spends billions of dollars on software research, purchases and implementation. If it marshaled those resources in support of open source solutions, it would achieve not only many of the clear advantages open source software delivers but would undermine the Microsoft monopoly at the same time. If the government demands uniform standards for Linux and other open source software for government purchases, this will go a long way toward preventing fragmentation of standards throughout the open source universe.

Many critics of the Microsoft suit raise reasonable concerns that a purely negative, restrictive approach to punishing Microsoft might inhibit innovation at the company without necessarily creating a viable competitor. Promoting open source software is the positive policy option that the government should employ to encourage the sort of innovation and competition that is needed to truly end the Microsoft monopoly.