Title: The Software Revolution

Course: EMGT 520
Term: Fall
Year: 1996
Author(s): S. Beers, A. Cordle, A. Coskun, L. Inman, H. Kassim and D. Kuran

Report No: P96052

Abstract: Analyzes the issues relating to the expansion of the Internet and the proliferation of the World Wide Web. The article suggests that both software and hardware as we know them today will transition to light weight programs that run on "information appliances."

The Software Revolution

S. Beers, A. Cordle, A. Coskun, L. Inman, H. Kassim, D. Kuran

EMP-9652

A

STRATEGIC AND POLICY ISSUES IN

ENGINEERING AND TECHNOLOGY MANAGEMENT:

THE SOFTWARE REVOLUTION

Prepared for:

Professor D. Kocaoglu
Engineering Management 520
Management of Engineering and Technology
Fall 1996

Prepared by:

Team #6
Scott Beers, Amy Cordle, Aysun Coskun,
Lane Inman, Hafiz Kassim, and Dogus Kuran

December 2, 1996

Table of Contents

List of Tables

List of Figures

1. Introduction

The catalyst for this strategic and policy issue analysis is the Business Week Article "The Software Revolution: The Web Changes Everything." [1] The article deals with a wide range of issues relating to the expansion of the Internet (Net) and the proliferation of the World Wide Web (Web). Specifically, this article contemplates the end of both software and hardware as we know them today, and the emergence of lightweight new programs that run on "information appliances." Only time will tell the extent to which these predictions come true.

What is more certain is that the growing phenomenon of the World Wide Web will continue to affect the strategic planning of information technology producers and users. This paper deals with the strategic and policy issues faced by five such groups as outlined below:

The Hardware Producer	What are the driving forces? What new options are emerging?
The Software Producer	How will emerging standards and new program paradigms affect development?
The Distribution Channel	How will the Internet and the Web affect software distribution?
The Corporate User	How will the Web, Java, and the network computer affect their information technology planning?
The Personal User	The personal user votes with dollars, what factors might they consider?

2. Historical Perspective

The Internet revolution began in 1969 as a project sponsored by the United States Department of Defense, Advanced Research Projects Agency (ARPA). ARPANET, as it was then called, was designed to permit computer communications over a wide area. By the mid 1970s, the basic services for remote connectivity, file transfer, and electronic mail (email) appeared. Many educational institutions around the United States were permitted to access the network to exchange research information. In 1990, ARPANET was taken over by the National Science Foundation. Networking companies and organizations providing the data connections to all the Internet host sites stepped up their efforts to provide easy global network access. [2]

The World Wide Web was created in 1989 by Tim Berners-Lee at the European Laboratory for Particle Physics. [3] It linked documents together through the use of hypertext, a proposed standard allowing troublefree cross-referencing.. Selecting hyperlinked text with a pointing device was all the user needed to do to access the information. In 1993, Marc Andreessen, co-founder of Netscape, created Mosaic with the help of fellow students at the University of Illinois, National Center for Supercomputing Applications. This graphical user interface simplified Web use. Since then, the use of the Web for information and commerce has grown at an exponential rate (See Figure 1) .

The Business Week article underscores how the Web and its browser software have ignited a personal computing revolution. The move is on to create a new wave of network-based software, hardware, and services that will exploit the Web and deliver the vision of the information appliance. The challenge has produced a surge of innovation and has created fresh opportunities for established and new players alike.

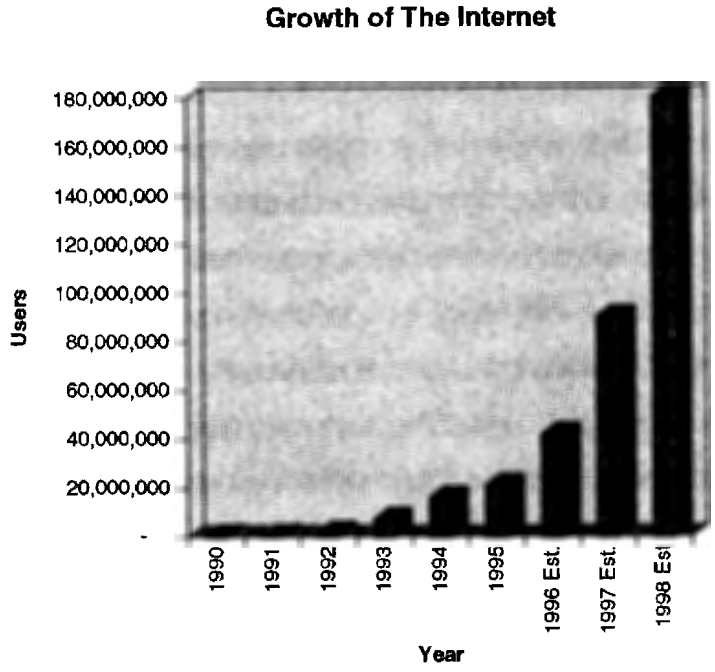

Figure 1 (Source: IDC) [3]

3. Hardware Implications

A significant result of the software revolution is an emergence of simpler hardware that offer a means of accessing the Internet. Development is occurring so rapidly that names are still being invented for the machine—network computer (NC), Internet appliance, Web PC, Java terminal, Internet access device, browser box, and Net-Top box. Whatever the name used, these are all minor derivations of the same concept. [4]

3.1 The Driving Forces

The NC concept is an inexpensive computer that discards today's overweight operating systems (OS) and bloated, platform-specific applications. Instead, it runs a microkernel OS and platform-independent software written in an interpreted language such as Sun Microsystem's Java. Monolithic, kitchen-sink applications give way to component-based programs and applets that you download from a network or the Internet only when you need them. In theory, the

NC would provide all the advantages of a PC with much less expensive hardware and software. [4]

Software would be more efficient because the user would load only the part of the program that is needed, when it is needed. For example, the Java applet could be a small (27 KB) spreadsheet. It would not perform linear regression or offer the extensive graphics of Microsoft Excel. The idea of components is that complex mathematics and graphics would be separate modules that the user could download when the modules were needed. Even at 28.8 Kbps, it would take about seven seconds to download such a spreadsheet. It would be much faster than that on a corporate LAN.

Although component-ware was invented for conventional PCs, NCs could benefit from it even more. NC proponents envision hardware designed for a distributed environment in which administrators store and maintain the software components on a network (server), not on the individual computer. The network could be a corporate LAN, an enterprise WAN, the Internet, or a secure Intranet.

Corporations are the first target for NCs. There are two reasons for this: they already have high-speed networks in place, and the administration costs for conventional PCs are soaring. Administration costs have become so significant, the low purchase cost of NCs would be an added, but secondary, attraction for corporations. According to studies by the Gartner Group, the five year corporate cost of owning a Windows PC is about $40,000. [4] By moving software maintenance to servers and preventing individual users from creating problems on their PCs, NCs could be easier for MIS departments to maintain and less susceptible to user errors. Also, this would prevent the use of unauthorized or illegal versions of software since new software would have to be installed by the MIS department.

NCs also could save money by slowing the costly spiral of upgrades. Each new release of a giant business package, such as Microsoft Office, requires more memory, more hard drive space, and more CPU cycles. More powerful computers in turn spur developers to create even larger software. If programmers broke up monolithic applications into smaller, dynamically-loaded components, tasks could be accomplished more efficiently.

The second major market for NCs is schools. Schools need more computers to prepare their students for a job market in which PCs are becoming as common as telephones, yet many schools cannot afford the high cost of traditional PCs.

Finally, there is the consumer market. There are no estimates of the number of people who would be interested in buying a home computer if it cost only $500 and gave them access to the increasing services on the Internet. The potential is so profitable, however, that at least a dozen companies, including Asian corporations that specialize in high-volume manufacturing of consumer electronics products, are pursuing the idea. [4]

Some observers have shown contempt for the idea that vendors can build and profitably sell a useful computer for only $500. Yet the industry sold millions of sub-$500 computers in the early 1980s, before the IBM PC standard raised expectations and prices. Perhaps the best selling personal computer of past times was the Commodore 64, which made its debut in 1982 at $595 and soon fell to $200. Today, a 486/33 with a small hard drive and minimal RAM could probably sell for about $500. Such a system might represent a low-end NC. [5] Computer technology has come a long way over the past decade, and it is now possible to build surprisingly powerful boxes at low prices. The question is not whether $500 computers are possible; it is whether a $500 computer can meet the expectations of users. Those expectations vary widely, depending upon the market and the application.

Mobile PCs will prove more resistant to obsolescence than stationary PCs because it will take time for wireless networks to become prodigious and fast enough to make mobile NCs practical. Eventually, however, public NCs could become as widespread as public phones. Instead of carrying a notebook and a bag of accessories, all one would need is a credit card. After the user logged on to the network, from the airport or hotel room, the NC would download the working environment and all the user's files.

3.2 Key Technologies

To transform the theory into reality, NC vendors must capitalize on four key technologies: cheap, fast micro-processors; lightweight system software and component-ware; platform-independent programming languages; and fast, affordable access to networks. Table 1 outlines these technologies and the products associated with them.

NC Enablers	Products to Watch
Fast, affordable access to networks	Cable modems ISDN ATM
Fast, affordable microprocessors	IDT R4640 NEC VR4300 Ultra-64 (Nintendo and SGI) StrongARM Superchips
Platform-independent programming languages	Java, Java Script Visual Basic, VB Script
Lightweight OSs	NCOS (Oracle) RISCOS (Acorn) GEOS (Geoworks)

Table 1. The Four Key Technologies (Source: BYTE, March 1996)

The software section, Section 4, deals with the issue of platform-independent programming languages and their implications. Following is a brief discussion of the technologies described in Table 1.

Fast, Affordable Access to Networks

The four technologies required to make NCs practical are all important, but networking is the keystone that supports the entire structure. Corporate users are the best prepared to adopt NC-centric computing models because they already have high speed networks in place. Depending on the network configuration and protocol use, considerable expense may still be involved to allow a corporation to run NCs on their corporate network.

The home user has greater difficulty obtaining fast, reliable network access. The World Wide Web, which barely existed five years ago, is one of the most remarkable networking and telecommunication developments for NC proponents. The Web and the Internet enable network-centric computing on a global scale. However, only a few million users with high-end PCs currently have access to the Web. On top of this, increasing Internet usage is bringing the phone systems close to gridlock by tying up millions of local phone lines every evening. [6]

There are several ideas for increasing the access capabilities of the average home user beyond the 28.8 KBps range of the most common high-end home modems. These include ISDN, Cable Modems, and Asynchronous Transfer Mode (ATM). All of these have infrastructure problems associated with them. The phone wires are all connected to analog switches and the cable industry is set up to communicate in only one direction, to your house. A significant investment in infrastructure, or a different way of using analog lines, will have to be developed to break this deadlock.

Fast, Affordable Microprocessors

For NCs to seriously challenge PCs, they must deliver comparable user experience and performance. That is not easy for a product that retails for $500 or even $1,000. A 120-Mhz Pentium chip alone costs about $350, and some versions of the Pentium Pro cost nearly $2,000. NCs do not need leading-edge microprocessors. Instead, NCs will ride the trailing edge of the price/

performance curve, where they can benefit from the changing prices to the full extent. The industry focuses so much attention on state-of-the-art CPUs that one forgets how much processing power is available for just a few dollars.

LSI Logic recently announced a superchip, called the Internet on a Chip, that is intended for low-cost NCs. The Internet on a Chip is part of a wave of highly integrated superchips coming this year from suppliers like Chromatic Research, MicroUnity, Nvidia, Philips Semiconductors' Trimedia Product Group, and Vadem. [7] Intel is less likely to be a player in this market, because it prefers to sell more profitable cutting-edge microprocessors. Those profits pay for the billion-dollar foundries that Intel needs to maintain its lead in production capacity and process technology. [4]

Platform-independent Programming Languages

To allow universal acceptance and utility of programs, a general platform-independent programming language must be created. Much like the *86 machine language is today for the modern PC, the new programming languages must be consistent and compatible across platforms. In essence, they must insulate applications from the details of the OS and Hardware.

This presents the challenge to vendors of producing a low overhead translator to reduce the amount of processing overhead needed to use the new language. It also represents a large hurdle in the development of code for proposed languages. Java is currently the leader in this race, boosted recently by Microsoft's licensing of the language. Microsoft, however, has announced VB script, a Visual Basic subscript intended to compete with Netscape Navigator's JavaScript. Also an interpreted language, this could directly compete with Java if Microsoft so chose to spend the time necessary for massive porting. Whichever language which is chosen, it will need to be ported to hundreds of different hardware platforms.

Lightweight Operating Systems

One of the most crucial technical challenges will be finding a lightweight OS that will let NCs compete against Microsoft Windows. Although the OS does not matter in the sense that Java programs will run on any system that supports the Java engine, the OS must be small and fast enough to run within the reduced resources of a low-cost machine.

The NC is Oracle's long-touted design for an inexpensive Internet-based computer that would pull applications off the Internet rather than needing applications to be installed into the computer itself. [8] The OS Oracle is developing is called NCOS for its NC reference design. The company has released little information about this OS, except that is based on a microkernel that supposedly runs in considerably less than 1MB of memory.

3.3 The Major Participants

The NC concept is spreading through the computer industry like wildfire. Acorn Computer, Apple Computer, Geoworks, IBM, LSI Logic, Oracle, Silicon Graphics, Sun Microsystems, and other companies are developing network computers or licensing their technologies to other vendors.

Sun Microsystems recently announced its JavaStation, based partly on the NC architecture. Sun also is developing a miniature operating system of its own, ideal for running applets, called JavaOS. It will power Sun's JavaStation network computers. Sun will even hard-wire parts of the OS into the microprocessors it is creating for the machines. [9]

Apple Computer set out to design an interactive media appliance for homes and schools, but it may have created the first network computer instead. Apple's Pippin has a low cost system architecture based on the Power Mac. Apple has no plans to manufacture Pippin devices. Instead, it is licensing the architecture to other companies. Although Apple did not design Pippin to be the kind of NC

envisioned by Oracle, Sun Microsystems, IBM, and others, the architecture has the flexibility to do just that. [4]

Computers are no longer luxury items; they are survival tools that are still priced like luxury items. The technology exists to produce a low-priced computer that could serve millions of people, and NCs may prove to do just that.

4. Software Implications

The software revolution is creating an atmosphere of change that evolves on almost a day-to-day basis, with ever increasing stakes. It captures the small entrepreneurs and the industrial giants with equal ferocity. The driving force is markets crying for another option to the traditionally large, feature-overstuffed applications and operating systems that continually increase the demands made on hardware, thus perpetuating a continuous obsolescence of expensive and complicated computer systems.

The root of the software industry's contribution to this software revolution is based on the concept of a standard interface that everyone accepts. A standard interface would eliminate the need for software companies to interface with the individual hardware or operating systems and would provide a common set of tools such as ActiveX, Java, or another common programming scheme. It also marks the migration from traditional software companies' static offerings covering a wide gambit of situations to customized dynamic software that uses only what is needed when it is needed. Such software services would provide perpetually up-to-date software, assuring that the user always has the most recent version of any software they care to use.

The theory says this will work with incredible ease; systems will be completely compatible with each other. Removing the requirement that software be coded for a particular hardware platform and operating system, software designers

would be free to design for all computers at once rather than for each individual platform.

Many of the decisions that must be made are based on standards that do not yet exist and will be determined through competition, cooperation, and compliance. To succeed a company must become a trend setter in one of two fields, either shaping the standards or following them. Each of these fields requires completely different approaches. [10]

4.1 Shaping the Standards

The software giants are best positioned at this time to capitalize on the renaissance of the Web and reap the standards benefits. The gamble that is being played out now is that current development will be supported by future profits, and those future profits are tied to technology that is being given away today. Netscape, who formerly held 90 percent of the browser market based on this concept, has lost 15 percent to Microsoft, who now holds 25 percent. Microsoft's strategy has been the very same; give away the browser software to lure developers towards ActiveX and the Microsoft back end server products. Sun Microsystems spun off Javasoft, "whose purpose isn't so much to make money from Java as to keep Java licensees marching down the same path." [11]

The trend setters of the past captured market share by pitting their rigid proprietary standards against those of competing companies, resulting in only one winner. The trend setters of the future must focus on generating a standard that everyone agrees on, quickly, and will buy into. The next step is to allow inexpensive access to the standard's software, allowing for quick, widespread development. This method provides a future revenue stream, much like AT&T's UNIX. If developers do not offer the standard software (Java, ActiveX, etc.) for free or very inexpensively, customers would be much slower to investigate and adopt the new technology. For trend setters, the money is in the future, not the present, and it is often through indirect channels. Sun, for example, has made

little directly from Java, but their revenues grew 25 percent largely due to increased sales of server hardware. [11]

4.2 The Trend Followers

For those who wish to build on the foundation laid out by the trend setters, the software market is extremely different. Using the now rather universal approach of HTML, companies may focus on the presentation of information more than the actual creation of new software. With highly available, call-when-needed "applets," software companies will need to determine whether they wish to provide server-side services or client-side software.

The corporate enterprise implementation model of this new software paradigm may require many tasks to be moved back to the server, using a universal "graphical" interface instead of traditional ones. Many traditional software service companies have jumped on the Java bandwagon and are doing just that. One such company is SAS, who is developing applets that let users access data warehouses and decision support systems. [12]

These database access methods use both server side cgi scripts and server/client Java applets. Much of this "software" is actual presentation of information contained in the database. The browser from Netscape will provide a common user interface to the database through the Java and cgi scripts. The money will not be in these front end components. As we have already observed, Netscape and Microsoft are giving their browsers away for free, or for very little money. It will be the software that runs on the server side interfacing to databases and legacy systems that earns software manufacturers money.

For users wishing to go beyond database access, additional applets or "plug-ins" enable the customer to add functionality and to customize their environment or application. These applets will be available for an extra fee, and will add to the complexity of the users environment with a corresponding increase in support

costs. The design and implementation of these applets will be an area of intense competition among software vendors, just as the 3270 emulator market is today. Since the plug-ins and applets can coexist, it will be possible for the user to mix and match feature applets from competing companies to finely tune their environment. This option is infinitely preferable to choosing between two or more monolithic software packages, neither of which actually suits all needs, and waiting until the next upgrade period to find out if the new version is going to contain the needed features or if it will be necessary to switch to a competing product. [13]

Eagerly awaiting the end of ever increasing operating systems, application suites, and feature-bloated software applications, many people see the technology embodied in the Internet, the Web, and Java-like languages as the solution to several problems. Since the home user market has a bottleneck at the modem, software companies will probably want to focus on network applications for corporate enterprise systems because the corporations have the resources and the bandwidth to digest the new technologies and serve as a test bed for the new network software. As will be discussed later in this paper, the corporations also have a more pressing need. When modem speeds and protocols are up to par for today's multimedia webbing, the software providers will have cut their teeth on the corporate market and will be ready for the home user.

5. Distribution Implications

The issue of software distribution has two facets. First, how will the web affect the sale and distribution of traditional software titles and licenses; and second, how will "rentable" or "disposable" applets be used. [1]

5.1 Software Distribution

Electronic Software Distribution (ESD), ordering software online and having it downloaded into your computer with the option to try before you buy, is a hot

which applet users will be required to enter into licenses prior to using their applets. The exact method for these types of transactions is still being explored.

Jade.org has created a set of form licenses for its members to use for different classes of applets. The applets are divided into three classes: freelets, with no payment of any kind; sharelets, with a small onetime or annual payment by users; and vendlets, or pay-per-view applets, for business use, which carry per-use charges. The new organization then helps its members collect any relevant fees for sharelets or vendlets, and verifies applets and classes being included in member applets. The group currently is seeking developer members, as well as VARs, integrators, and consultants. [16]

Arthur van Hoff, one of the original members of the team that developed Java and Hot Java at Sun Microsystems Inc., said although he thinks the jade.org idea is intriguing, he is not sure if it will work. "I'm of the opinion that user software will remain free while services will cost money," he said. [16]

6. Corporate User Implications

To date, the Java applications most of us have seen have been forms on Web pages. However, for Java applications and NCs to really excel in the corporate environment, strategic business applications are needed to drive acceptance. According to a survey of Fortune 1000 companies by Forrester Research, 62 percent already use Java for some development, and 42 percent expect Java to play a strategic role in their company in the coming year. [17]

According to Sun Microsystems, the driving forces for adopting Java-type languages and NCs include the following [17]:

- The realization of true open computing for the first time.
- Sharply reduced cost of client administration, as well as big savings in total cost of ownership.

- Rapid development and deployment of new computing applications.
- Fuller use of existing legacy systems.
- Improved security.

One of the key strategic applications corporations are looking forward to is the availability of a truly open system and the emergence of the "extranet". With the common platform promise of Java and the Web, corporations will be able to use the same software for collaborating and communicating outside the firewall as inside the firewall. Communication and information exchange with business partners, customers, and vendors will be much simpler, and the process associated costs will drop dramatically. [18] James Barksdale, CEO and president of Netscape and formerly with Federal Express, had this to say:

We used to build these nice little tight proprietary systems, spend a lot of money on them, and we'd never think, "Well, maybe I want another constituency to look at this some day." At FedEx, we would have killed for this kind of technology 15 years ago when we built some of these systems. [18]

We have already discussed the potential cost savings associated with the Java/NC model. Sun claims a yearly cost savings of around 75 percent over similar PC applications. [17] This is difficult to apply corporate wide, however, since many of the applications that are available on PCs are not yet available on the Java/NC platform.

The general corporate feelings about the Java\NC platform are shown in Figure 2.

The general corporate feelings about the Java\NC platform are shown in Figure 2.

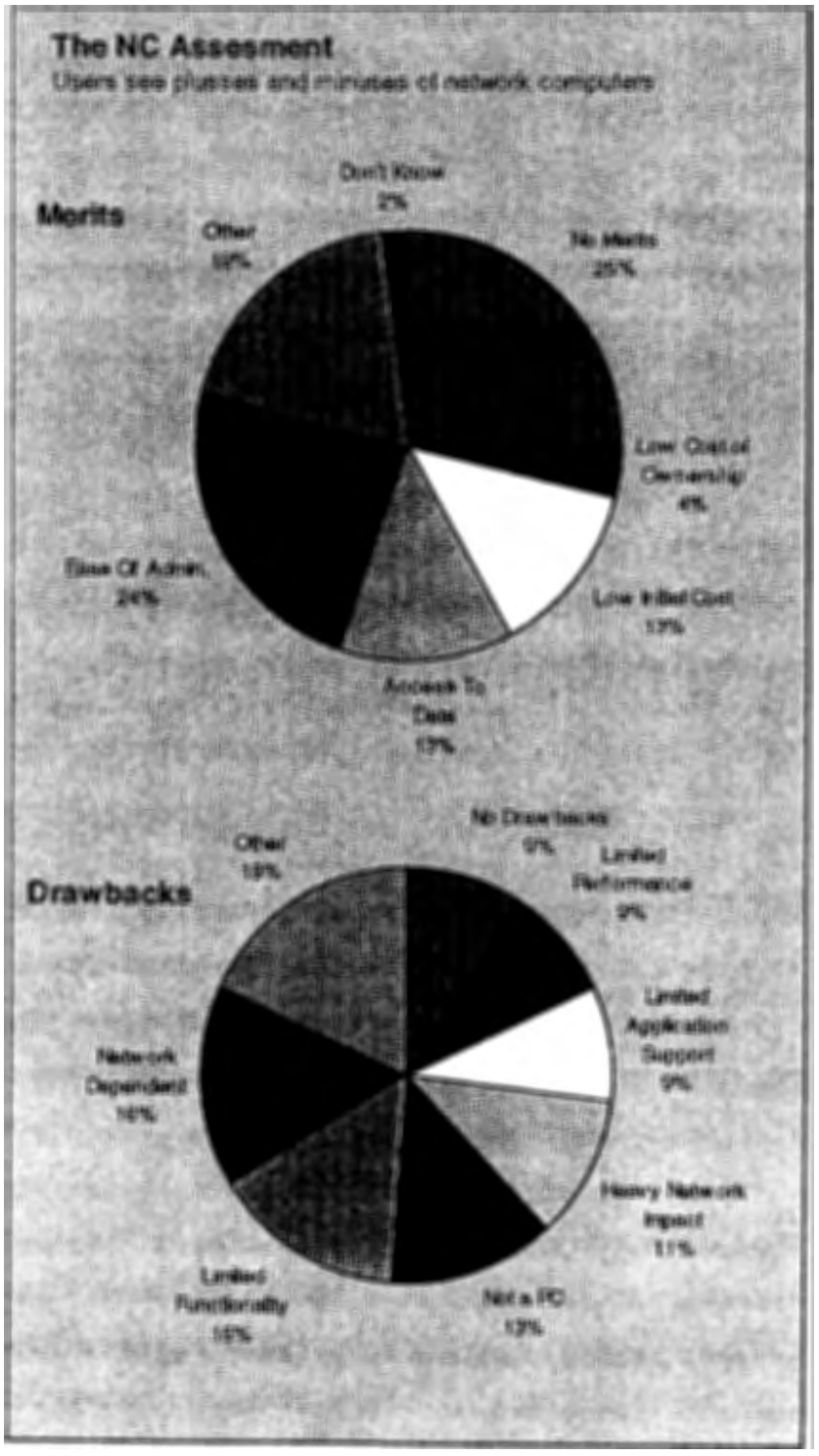

Figure 2 (Data: Zona Research survey of 55 user sites, half of which have more than 1,000 employees [19])

The average Fortune 1000 corporate computer user usually has access to the following software:

- Word Processor
- Spreadsheet
- Database
- Group Calendar and Scheduling
- Email
- Mainframe Access
- Client/Server Access

Depending on that user's job function, they also may have access to the following applications that often run in specialized environments (client/server, mainframe, high speed workstations):

- Accounting Applications
- Purchasing Applications
- Inventory Applications
- Manufacturing Scheduling Applications
- CAD/CAM/CIM Applications
- Shipping Applications
- Sales Force Automation
- Company Intranet

Given the breadth of applications in use, and the relative scarcity of Java applications, corporations will probably not be completely switching over to NCs and Java soon. It is a good vision, with powerful driving forces, but the infrastructure for distributing, managing, and using that model is not presently there. [19]

7. Personal User Implications

According to the Business Week article, personal users generally would stand to benefit from the predicted software revolution. The greatest benefits would be monetary—personal users would no longer require the high-priced software of today to perform needed functions or the hardware to support it. Personal users also would not be subject to the need to upgrade their computers every few years to support gluttonous software packages. Web-compatible software

languages would mask computer platform restrictions, so personal users would not be limited in their choice of purchasing a PC or a Mac. [1]

Several such scenarios are presented in the article. Rather than purchasing high-priced software, personal users would be able to access, via the Internet, applications to perform just the needed tasks. For instance, the personal user could download a tool for balancing a checkbook or filling out a tax form. And instead of purchasing upgrade after upgrade of their chosen software, the personal user may be able to obtain subscriptions for different software. In essence, the user would be able to rent or lease the right to use a given software package. One outcome of these software changes would be the elimination of the need for ever-faster personal computers with ever-increasing memory since software and data would be stored remotely. However, with increased Internet and server reliance would come an increased need for better modem and transfer technology and a more reliable Internet infrastructure.

While these changes portend great things for purchasers of computers and software, they overlook the fact that many personal home computer users, including small business users, have chosen to upgrade computers and software every few years. To use the new network appliances, the user will have to throw away their existing operating system and applications and buy or rent downloadable online applications, raising the investment cost above the $500 to $750 price tag being touted by Network PC supporters. By taking the processing function out of the PC, power and flexibility would be shifted away from the personal user and back to a distant mainframe. [20] The result, given current and reasonably foreseeable communications technology, would be a reduced performance over what personal users are accustomed to. [21] The graphical nature of many of today's applications (e.g., icons in windows-based software) would only further reduce performance. Many personal users are unlikely to relinquish their high-priced machines to small Internet appliances or to give over control of their software and personal files to a service in the foreseeable future. [22]

While the software revolution and availability of Network PCs may be less appealing to existing personal computer users, there is a large potential market of consumers who do not yet own personal computers. The lower initial investment over traditional PCs is an attractive alternative for those who cannot afford today's PCs or choose not to make the investment. The ability to use the PC without significant software investment would add to the appeal. The Network PC would be particularly useful to consumers whose main interest is accessing the Internet and having Email capabilities. For instance, one potential market may be parents and grandparents whose main interest is communicating with their children and grandchildren. Simpler software and hardware also would appeal to those who have never used a computer or have no means of easily learning how to use a computer.

The great number of companies interested in the new Internet-driven market offers good consumer opportunities in that competition often breeds increased quality and decreased costs. It also offers an unstable arena, however, while companies explore what the market will handle.

8. Conclusions

If we could predict what the market was going to do, and when, we could all retire to a warm sunny climate and pay someone to worry about any attendant tax problems. It seems safe, however to make some predictions about who will adopt what technology, and why.

When faced with a choice of emerging technology, especially in an arena as complex as the one described herein, there will be early adopters and late adopters. Both the software and hardware industries will have to deal with a variety of combinations and permutations of old and new hardware and software technologies until a clearer standard or direction emerges. In the short term, we see the following technology adoption groups:

Old Software and Old Hardware Technology

These are the people who can't afford to, don't understand why, or don't need to upgrade. They are the late adopters in a technology implementation curve.

Old Hardware, Old and New Software

There are a variety of people in this class including the following:

- Corporate users that need to maintain backward compatibility with older programs that are not available in the new format, or are too expensive to convert to the new format. They will implement the new software paradigm where it makes business sense to do so, but are still tied to the old platforms via their need to run old style software.

- There are some software products that may always run in the old "Workstation" environment. CAD/CAM programs are an example of software that may be workstation-tied for some time to come.

- Users who require local storage for convenience or security also will opt for their older equipment. It is unlikely that anyone is going to let their Quicken home accounting data be stored on the hard drive of their Internet ISP any time soon. Corporate users who need laptop access, and who load old style software ,often also will require local storage.

New Hardware, New Software

There will likely be three classes of users who use both the new hardware and new software platforms:

1. The corporate user who has converted all operations to the new paradigm.
2. The home user that uses the Internet for access to information and is content with storing any data files with an Internet ISP.

3. The industrial/commercial user who uses the new hardware and software in a very specific way to deliver a turnkey package of services.

Suffice to say that there will be quite a blend of requirements for some time to come as we transition to this new paradigm defined by the Web, the Net, and Java.

The concept of the "Network Computer" is not a new idea. It may be, however, an idea whose time has come. Technology users are growing increasingly unsatisfied with the status quo, caught in an expensive spiral where hardware upgrades chase bloated software. As the computer transitions from a luxury toy to a necessary tool, market forces will shape new products that will accommodate the ever-increasing variety of user needs and market demands. Though no one can say for certain what these products will finally look like, a great number of people are wagering large sums of money that they have an answer that is close enough.

9. Bibliography

[1] Cortese, Amy, et al., "The Software Revolution: The Web Changes Everything", Business Week, December 4, 1996, pp. 78-90

[2] C. Hunt, TCP/IP Network Administration , USA: O'Reilly & Associates, Inc. 1994, pp. 5-7.

[3] Hills, Melanie, Intranet Business Strategies, John Wiley & Sons, 1996

[4] Halfhill, Tom R., "Inside the Web PC", BYTE, March 1996, pp. 44-56

[5] Sager I. and P. Elstrom, "The New Electronic Hearth", Business Week, November 4, 96, pp. 116-126

[6] URL http://www.cnn.com/TECH/9610/29/gridlock.reut/index.html

[7] Unknown, "News Brief", *BYTE* , November 1995

[8] URL, http://biz.yahoo.com/bin/jump?/financial/11/04/orcl_sunw_1.html

[9] Schlender, B., " Sun's Java: the threat to Microsoft is Real", Fortune, November 11,1996, pp.165-170

[10] Miller, Michael J., "The Next Software Revolution", *PC Magazine*, March 30, 1993 v12 n6 p81(2).

[11] Schlender, Brent, "Sun's Java, The Threat to Microsoft is Real", *Fortune,* November 11, 1996

[12] Cunningham, Cara and Smalley Bowen, Ted, "Java Apps Will Push NC Acceptance"; *INFOWORLD*; November 4, 1996.

[13] Thompson, John M., "Hot Seat: Reinventing IBM", *INFOWORLD*, October 7, 1996; page 27.

[14] Young, Monica, "Electronic Distribution: Friend or Foe?", *Computer Reseller News*, October 28, 1996.

[15] Young, Monica, "Experts offer checklist for building an effective, secure infrastructure – More VARs Navigating Maze of ESD Do's and Don'ts", Computer Reseller News, November 4, 1996.

[16] Taft, Darryl K., "Java group focuses on applet license plans", *Computer Reseller News*, March 4, 1996.

[17] Unknown, "Java Computing in the Enterprise - Strategic Overview", *Sun Microsystems White Paper*, August, 1996

[18] Unknown, "Beyond Browsers - An interview with James Barksdale", Information Week, October 7, 1996.

[19] Hayes, Mary, "The NC Arrives", Information Week, Nov 18, 1996, pp. 14

[20] Fred Langa, "Dumbed-down "thin" Network PCs? Thanks, But No Thanks," *Windows Magazine*, February 1996

[21] Unknown, "Back to the Garage: the best software revolutions start small," *The Economist*, May 25, 1996

[22] Lewis, Peter, "Doubts About the Fantasy of a $500 Network PC", *New York Times*, November 27, 1995