

<b>Chapter 2.4 History of the Internet</b>	<b>1</b>
<b>The Cold War and a balance of terror</b>	<b>1</b>
Sputnik upsets the balance and ARPA is created	1
RAND, Paul Baran, and packets	2
<b>ARPANET will use packets</b>	<b>3</b>
How to handle the packets?	3
BBN gets the contract to build ARPANET	4
<b>ARPANET to INTERNET</b>	<b>5</b>
AlohaNet	5
SATNET	6
PRNET	6
<b>Military to academic to commercial</b>	<b>6</b>
CSNET	6
MILNET	6
NSFNET	7
<b>More members of the network zoo</b>	<b>7</b>
Batch is replaced by time share	8
The personal computer	8
Dial-up bulletin boards	9
Local Area Networks (LANs)	10
Why the LANs were important for the Internet	10
<b>Applications for the Internet</b>	<b>10</b>
telnet	10
ftp	11
e-mail	11
World Wide Web	11
Why is Internet Explorer free?	13
JAVA	14
Microsoft invents C# and .NET	14
Distributed computing	15
<b>The future of the Internet</b>	<b>15</b>
Figure INTimeLine	17
Internet time line. Events associated with development of the Internet are arranged chronologically:	<b>18</b>

## Chapter 2.4 History of the Internet

### The Cold War and a balance of terror

The pace of military research and development was intense during World War II, and superior technology was a significant factor in the victory of the Allies. Inventions from antibiotics to radar changed both military and civilian life during and after the war. However, the invention of the atomic (fission) bomb and its use in 1945 to win the war against Japan initiated an entirely new era, and established the United States as the dominant military power in the world. While it quickly became clear that the Soviet Union would play a unfriendly and competitive role in the post war era, it only became a real threat to the US after it conducted a test explosion of an atomic bomb in 1949. Test explosions of hydrogen (fusion) bombs by the United States and then the Soviet Union in 1953, ratcheted up the tension, since these were about 1000 times as powerful as the original atomic bombs.

But not to worry, the United States was on the other side of the earth from the Soviet Union, and it would take many hours for bombers to go the distance. During those hours they would be attacked by swarms of fast jet fighters launched from Europe, Japan, and Alaska, even before they were in range of mainland America. Meanwhile United States bombers would be on their way to the Soviet Union carrying enough nuclear explosives to convert the Soviet Union (and perhaps the entire Earth) into a glass wasteland. Not exactly a peaceable kingdom, but it seemed that even a madman would hesitate to upset this balance of terror.

### Sputnik upsets the balance and ARPA is created

In October, 1957 the USSR launched Sputnik, a basketball-sized satellite that orbited the earth transmitting a Morse code message in the short wave radio band. Of course Sputnik generated great interest in the scientific community. I was an undergraduate at Caltech at the time. When the existence of Sputnik was announced by commercial broadcasters we immediately tuned short wave receivers to find the signal. The receivers were then connected to chart recorders to record the amplitude and frequency of the audio signal. After each orbit of the satellite, about 90 minutes, the chart paper would be torn off and taped to the wall of the hall. Measurement of the Doppler frequency shift as Sputnik passed over us revealed its speed, and thus confirmed that it was indeed orbiting several hundred miles above the earth as reported by the Soviet authorities.

Sputnik posed no immediate military threat, but it was a dramatic technological accomplishment in comparison to the more modest performance of US rockets. For the first time the USSR had demonstrated dramatic technological superiority to the US. Was it just luck, or was there a series of progressively more powerful rockets just waiting to be tested? President Eisenhower felt the need for military intelligence gathering to determine the size and nature of Soviet air and rocket power and high level technical advice to plan an innovative response to the Soviet threat. The need for

intelligence lead to the high altitude U-2 photo reconnaissance plane that could fly over the Soviet Union, above the range of anti-aircraft rockets (above at least most of the rockets). The need for innovative research and development resulted in the creation of the Advanced Research Program Agency, ARPA, in 1958 (the name was changed to Defense Advanced Program Agency, DARPA, in 1972, changed back to ARPA in 1993, and back to DARPA in 1996). Because of his experience as Supreme Allied Commander in WWII, Eisenhower insisted that these new intelligence and research programs be independent of the military branches. ARPA gave grants to university groups in order to perform defense related studies, and the fact that some of these funds were used to purchase computers will eventually get ARPA into the business of funding the precursor to the Internet.

#### RAND, Paul Baran, and packets

The freeway going West from Los Angeles to Santa Monica dead ends only a few hundred yards from the Pacific Ocean. On the left is a brick and concrete cubic building, unadorned by any obvious identification. This is the home of the RAND Corporation, where we will start the history of the Internet. RAND (an acronym for Research ANd Development) started as a project in the Army Air Force in 1946, immediately after the end of World War II. Two years later it was spun off as a non-profit corporation supported by military contracts, mainly in the areas of operations research and computer technology.

Paul Baran was at RAND in 1960 when he wrote an extensive series of proposals and specifications for a digital communication network in which data would be sent to their destination by autonomous routers at nodes of a network. The goal was to create a communication system that was resistant to nuclear attack. If multiple nodes routed data through multiple links, there would be no single switch or link that would be vulnerable to an attack.

The system imagined by Baran was is a store-and-forward network. A node receiving a message does not know the link the message will be sent out on, since a complete sender to receiver path, or circuit, has not been defined when the message was sent. Rather, the node must store the message while it reads the destination address from the message and then determines the best route to send it on its way to the ultimate destination. Baran was not the first to propose a store-and-forward network, but his insight was to realize that the network could only be efficient if the nodes handled very short messages. Then the nodes don't need a large memory for long messages and could handle input from multiple nodes without large delays. Of course, some messages would actually be long, but you then just cut them up into many short packets, and reassemble the packets at the destination. This creates extra work, but Baran insisted it was worth it. In order to implement the packet system you need computers and software at the nodes. That would come later, but Baran gave many talks and published multiple articles on the advantages of the packet system, and thus created an environment for its creation.

**ARPANET will use packets**

Computers were an important tool for ARPA, since they could enable design of weapon systems, analysis of intelligence data, etc. The Information Processing Techniques Office, IPTO, was formed within ARPA in 1960 and funded timesharing computers at academic institutions doing contract work for ARPA. Of course, every group wanted their own computer, but the forecast for the price of this policy became alarming. An obvious solution would be for computers to be shared. However, there was no easy way for groups at different locations to share computers, even if they wanted to. In 1966 Robert Taylor became director of the IPTO, and hired Lawrence Roberts to create a network, ARPANET, to link computer sites funded by ARPA. In response to a rather informal proposal Roberts was given half a million dollars in 1967, and 2.2 million the next year to start ARPANET.

The goal for ARPANET was to not to just allow one user at a keyboard to send characters to one remote computer, but rather to enable many users to transfer large computer programs to and receive large data files from many computers. This would require leased high speed telephone lines (high speed for the 1960s meant 56kbs) and robust error correction routines. Data transfer to and from computers consists of short periods of activity and long period of silence, thus many users could share one line if there was a method of addressing and switching data from sources to destinations.

In 1966 Donald Davies at the National Physical Laboratory (NPL) in London had proposed a packet switching scheme exactly for the purpose of linking time sharing computers, and by 1967 the network was under construction. In October of 1967 Roberts attended a computer symposium in Gatlinburg, Tennessee where he learned of the nascent NPL network and heard several participants suggest a packet network as the clear choice for ARPANET. Soon afterwards Roberts read Baran's report, and then made Baran an advisor for the design of ARPANET. Resistance to nuclear bombs was not part of the design specs for ARPANET, but many of the basic ideas of Baran were used in its design. Now ARPANET was committed to be a packet network.

**How to handle the packets?**

The decision to use packets on ARPANET was ambitious and radical. No functional packet network was running at that time, so creating ARPANET wasn't just modifying an existing system. There were certainly many details to be worked out, but a major problem was deciding how the data manipulations needed for a packet network were to be done. Since this was a network linking mainframe computers, you could propose the mainframe computers would do all the work. In fact, only two years earlier, the only alternative would have been to build dedicated hardware to perform the packet manipulations. However, dedicated hardware would have been expensive, and you can't build hardware until you know the exact procedure; if you change the procedure, you rebuild the hardware. One of the important advantages of the programmable computer is that the program (procedure) can be changed by merely writing new code and loading it into the computer. Since the ARPANET packet network was essentially an experiment, the structure of the packets and procedures for handling them would certainly change many times in its development.

In 1965 the Digital Equipment Corporation had produced the first mini-computer, the PDP-8, which cost about \$20,000 and was the size of a desk. Wesley Clark, one of the ARPA Principal Investigators (Pis), was influential in arguing that ARPANET should be implemented using mini-computers. At each node of ARPANET the same model minicomputer using the same program would implement all the network protocols. This minicomputer and software was named the Interface Message Processor, IMP. The mainframe computer would then exchange data with the IMP as if the IMP was a standard peripheral device. This approach had both political and engineering advantages.

The people that controlled the host computers were not sure they wanted others to use "their" machine, and others included those working on ARPANET. They were also skeptical that the packet network scheme would work (some of the people working on ARPANET weren't so sure either). Creating an independent network meant that the mainframe computer and associated staff didn't have to be disturbed during the Internet development process and the inevitable failures didn't have to be shared with a hostile audience. Thus there were strong political and psychological advantages of segregating the net related tasks to a separate computer system.

If the same model IMP was used at all computer centers as a gateway to ARPANET, only one software program needed to be written to run on that IMP. Once two IMPs could communicate with each other, it would be easy to expand the network by buying more minicomputers and running the same software on them. To be sure, there might be small differences in the way the IMPs communicated with different host computers, but since input and output devices made by different manufacturers were often mixed, incompatibilities were a very minor problem compared to writing complex, experimental packet handling software for multiple machines. These were the engineering advantages of using standard minicomputers to implement the network. Years later, when packet software was fully developed and everyone wanted to use the Internet, the software became a standard package distributed as part of the operating system on all computers.

#### BBN gets the contract to build ARPANET

In the fall of 1968 ARPA published a Request for a Proposal (RFP) to build the IMPs for ARPANET. IBM and Control Data Corporation, the two biggest computer companies at the time, refused to submit a bid, saying it couldn't be done within any reasonable budget constraints. Only Raytheon and a tiny company named BBN submitted a final bid. The contract was awarded to BBN, Bolt, Beranek & Newman, a consulting group in Boston founded by MIT graduates to use modern acoustic theory to design symphony halls etc. Computers at BBN were at first an internal design tool, but they gradually developed a small software consulting business. BBN got the contract to develop the IMP because they actually believed packet switching might work, and they very much wanted to be the ones to do the job. The BBN proposal was very detailed, outlining all the likely problems and solutions. Finely, it didn't hurt that several of the engineers at BBN were classmates of Roberts.

ARPA had assumed the PDP-8 would be used as the IMP, since it was the first and best known mini-computer. However, BBN picked the DDP-516, a mini-computer

manufactured by Honeywell which had the appeal that it could be purchased as a "rugged" version that BBN thought would be more likely to survive the academic environment. The first IMP and the network software was delivered on schedule to UCLA in August of 1969. The machine was greeted at the loading dock by graduate students who thought the "real" network programmers would show up later, but gradually realized that they were the programmers. The second IMP was set up at The Stanford Research Institute, SRI, in Palo Alto, California, and messages were sent between UCLA and SRI in October. In the next two months an IMP was installed at the University of California at Santa Barbara, and another at the University of Utah at Salt Lake City. Thus, by the end of 1969 there was a working, four node ARPANET; the BBN contract had been finished on schedule.

### **ARPANET to INTERNET**

The evolution of ARPANET to INTERNET was catalyzed by experience from several very different networks.

#### **AlohaNet**

Abramson at the University of Hawaii, frustrated by noisy telephone lines, created Alohanet in 1970, using radio transmission to send packets between computer centers. Since different centers shared the same radio frequency, the environment was very different from ARPANET where there was a committed telephone line between each pair of IMPs. When two computers on Alohanet sent packets that overlapped in time, and thus interfered with each other, the collision was detected and the two computers waited a random, and thus usually different time interval, and then retransmitted their packets. The second transmissions did not usually conflict, but if they did the process was repeated. The success of Alohanet proved that the individual links and nodes of a packet network could be shared by many users, as long as brief delays in transmission could be tolerated.

Alohanet not only contributed to the structure of the Internet, it also led to a successful commercial network system that could link several dozen desktop computers together to create a Local Area Network (LAN). Metcalf worked at Xerox PARC in 1972 and implemented the Alohanet collision and retransmission strategy using a group of computers linked by cable. He devised an improved algorithm for retransmission timing in his MIT Ph.D. thesis, and then founded 3Com, a company that sold the hardware and software to implement these ideas under the trade name Ethernet. starting in 1983. The rapid growth in the number of workstations, and later personal computers, thus generated a large number of isolated LANs. When the LANs were connected to each other in the late 1980s and 90s using the Internet, it resulted in an explosion of interconnected computers. While there are other local network protocols, Ethernet is simple, robust and inexpensive. It has gone through several cycles of upgrades to progressively higher speeds over three decades, and remains the predominant type of network for linking small computers together.

### SATNET

In 1973 the Alohanet was connected to University College, London, using the Intelstat I geostationary earth satellite. In the same year, Cerf and Kahn described a method for connecting the ARPA networks, and Cerf gave a lecture on a Transmission Control Protocol, TCP, which would enable reliable packet transfer on basically unreliable networks by re-transmitting packets that were not received and acknowledged by the target host computer.

### PRNET

In 1975 a test network of military radios was set up in the San Francisco that exchanged digital information as packets, just as if they were part of a computer network. This project was directed by Kahn, and the network, intended for use in combat, was called PRNET. Two years later information was exchanged between PRNET, ARPANET, and SATNET through gateways that translated information passing between networks using different protocols. This was a significant demonstration that an Internet could be constructed.

### **Military to academic to commercial**

ARPANET was ultimately a military project, since the computer sharing it was created to enable was intended to be used for defense projects. In principle only ARPA funded project personnel were to have access, and only ARPA related communication was to flow on ARPANET. However, the network was experimental and thus not used for critical communication. In addition, since it was underutilized, a considerable amount of unofficial, and even whimsical, data was exchanged on ARPANET; after all, the more traffic the easier it was to test and debug the system. However, ARPANET was not easily accessible to the entire academic, or even the entire academic computer science community, who gradually became aware that their colleague were using e-mail and remote computers while they were not.

### CSNET

Lawrence Landweber, chairman of the computer science department at the University of Wisconsin, initiated the effort to provide network access for computer scientists in meetings held on the campus in 1979. Vint Cerf suggested that it would be desirable to use the ARPANET packet formats, first because they worked and second because it would then be easy to communicate between the new network and ARPANET. In 1981 the National Science Foundation (NSF) contributed \$5 million to start the Computer Science Network (CSNET). Gateways between CSNET and foreign (non-US) networks were also set up to allow exchange of e-mail.

### MILNET

As the size and reliability of ARPANET grew the military community began to rely on it for real communication purposes. This expansion was sure to conflict with the increasing use of ARPANET and connecting networks by civilian and even foreign

sites. The conflict was averted in 1983 when MILNET, a separate network for use by the Department of Defense, was created.

### NSFNET

In 1987 NSF made a 5 year commitment to build a upgraded backbone to carry network traffic between regional networks. The leased lines used by ARPANET operated at only 56 kbps, and the increasing number of ARPA sites plus other users, like the NSF funded CSNET, were more than these lines could handle. MERIT was chosen to build and operated the new backbone, routers were supplied by IBM and the leased lines provided by MCI. Until NSFNET was completed ARPANET lines would carry the load, but NSF would share some of the cost. As portions of the NSF backbone became operational the ARPANET lines became redundant, and at the end of February 1990 the ARPANET lines were shut down for good. The Internet was now civilian, but not commercial.

### CIX

During the late 1980s a commercial market for regional IP networks slowly developed. The customers were Internet users that needed a low cost connection to the NSF backbone as well as private corporations that exchanged data between locations or provided a service to retail users. As these networks grew the need for a national network increased, and in 1991 three of the biggest, PSINet, CERFNet, and Altnet, agreed to exchange messages between themselves in order to increase the size of the region they could service. An important and unusual characteristic of this network of networks is that members exchange data not money; a network "pays" for transfer of messages to other networks by accepting messages from those networks. Clearly the implementation of this exchange requires constant adjustment and ingenuity, but note that the basic nature of routing on the Internet would make any alternative even more difficult since the route of a message on the Internet is not known when it is sent or received. It would require modification of Internet protocols and a massive book-keeping system to record the route of the portions of a message. Many of the messages sent over the Internet are short, e.g. a request for a web page or the acknowledgement that a web page has been received at its destination. As a result, the accounting overhead would be larger than the effort needed to transmit the message.

More regional networks joined CIX, and a world wide commercial system was created in just a few years. The NSF had been trying to transfer responsibility of running the Internet to the private sector for many years, and now it could do it. In April of 1995 federal funding for the NSFNET backbone was terminated, and all Internet traffic was carried by CIX. No longer were users required to adhere to the "Acceptable Use Policy" which forbade any communication with commercial motives.

### **More members of the network zoo**

ARPANET was created to enable communication between mainframe computers, or more accurately between IMPs that were each directly connected to a mainframe. We have described several other networks developed under ARPA and NSF

sponsorship, which when connected to ARPANET created an Internet. However, there were many other types of networks built concurrently, mostly civilian and commercial, that established an awareness of the usefulness of networks. When ARPANET matured and was freed from restrictions imposed by government funding, these other networks merged with the Internet to produce an exponential explosion in size.

Batch is replaced by time share

Early mainframe computers were single stream batch machines. A single program was run by the machine at any one time. As the machines became bigger and more popular they were used by more people, many who were not directly connected with actually running the machine. The program to be run was typically encoded on hundreds of punched cards which were submitted to an Input Clerk at the User Room in the Computer Center. A day later the Output Clerk would deposit your output from the computer into a mail box in the User Room. While this system seems medieval today, it had the positive effect of creating a community of computer users that met in the User Room and exchanged programming experiences and advice.

Users of a time share computer sit at terminals and enter programs directly into computer memory. Users then submit the program to be run, and the computer returns the output within minutes. In order to have a functional time share system there need to be sufficient terminals that all users can have access to one, and the operating system must support multiple users in a way that creates the impression that all of them have immediate access to the computer. This impression is created when the computer sequentially looks for input from all the terminals at a rate that obscures the fact that the computer can really only work on one job at a time. To be effective the time share computer must have excess capacity so that users are unlikely to experience a significant pause in response to input from their terminal. The goal for the time share machine is to maximize the efficiency of the user, while the design of a batch system was to maximize the efficiency of the computer.

Brian Kernighan started to write a time share operating system in 1970, and was joined by Dennis Ritchie three years later. These two people created the UNIX operating system which quickly became the most popular operating system at academic centers.

When a group of users are simultaneously using terminals connected to a time share machine you have a network. If the people use the machine frequently, and especially if the terminals are in individual offices, it becomes convent to exchange messages between users. If the user is on the terminal a message is displayed, and if the user is not using the terminal the message is just stored until the user logs on the terminal. E-mail is born!

The personal computer

The market for the first personal computer was electronic hobbyists and the first kit, the Altair 8800, started the industry in 1975. It was sold by a company that previously

made its living by selling radio controls for amateur rocket builders. Apple Computer sold the first fully assembled popular personal computer in 1976.

The business community became aware of the personal computer when the spread sheet application was created. A spread sheet is a table of numbers, where some of the numbers are related to others in the table by formulas. The revolutionary aspect of the spreadsheet is that when you change one number, all the other numbers in the spreadsheet that are related to that number by a formula immediately change. It's an example of transparent time sharing computing, where the user doesn't need to submit the new table to the computer for it to be updated, it just is updated.

The ability to modify any number in a set of complicated calculations and have the result immediately recalculated was too exciting for the new wave of MBAs making their mark in the business world. Thus, the spread sheet was the main reason that IBM granted a maverick group of engineers permission to develop the IBM PC, and the rest was history.

#### Dial-up bulletin boards

The first impact of PCs on network communication was the creation of a simple network called a bulletin board. The owner of a bulletin board maintained a computer connected to multiple modems, each connected to a telephone line with the same calling number. The computer was often a PC with a generous amount of disk space, or sometimes a small mini-computer. Users would dial up this host using a modem and PC. Typically a bulletin board user could leave and retrieve messages from other users (e-mail), exchange messages in real time with other users that were connected (instant messaging), and participate in discussions of topics (mail-lists). Users would pay the bulletin board owner modest monthly fees, thus creating a cottage industry. Bulletin boards were typically organized around specific interests, e.g. sex, sports, computer technology. They were advertised in computer magazines and throw-a-way newspapers. Most bulletin boards were not so compelling that many users would pay long distance telephone rates to link to them, and thus they were a network within one area code. However, some bulletin boards formed a network that exchanged e-mail at night when the long distance rates were lowest.

One of the more notable bulletin boards was The Well (The Whole Earth 'Lectronic Link'), founded in 1985 by Larry Brilliant of Networking Technology, and Stewart Brand of The Point Foundation. The Point Foundation also published The Whole Earth Catalog, which featured tools to enable the hippies of the 1960s and 70s to live by their hands and wits in rural locations. The host computer for The Well was located in a small warehouse at the end of Napa Street in Sausalito, on the waterfront of San Francisco Bay. Sausalito was home for hundreds of house boats and a large counter-culture community. The Well used software that supported many discussion groups, and the subjects of these groups reflected the culture of the area and time. The sense of community was enhanced by an annual party at The Well headquarters. The Well was one of the first non-academic sites to establish a gateway to the Internet in 1992, but for the first few years gateway users were required to promise that they would use the Internet for only educational, and not commercial purposes. The Well is still in

business (at least as of 2002), but of course you now reach it over the Internet, not by a direct dial up modem. The host computer is in San Francisco and the culture a little less radical. Sausalito is also less radical; the houseboats are now hooked up to the municipal sewer system and most require a substantial mortgage.

### Local Area Networks (LANs)

As PCs became part of the business office, it became desirable to link them together into a Local Area Network (LAN). At first the main motive was to share a printer or information from a central data base, but as the networks became larger e-mail became more popular. The workstation computers made by Sun, which used the Unix operating system, had TCP/IP network capabilities from the beginning. The Macintosh computers made by Apple also came with networking hardware and software, using a proprietary protocol, naturally called Apple Talk. Terminal emulation software allowed these desk top machines to be connected to a mainframe computer and exchange information as if it were a terminal.

### Why the LANs were important for the Internet

By the time packet switching had been developed on ARPANET and then morphed into TCP/IP and joined with NSFNET, millions of users of desk top computers had discovered the advantages of local area networks and their potential for communication. While many of these LANs were small and used a collection of incompatible protocols, people knew what networks were and used them. As the Internet became open to all academia, then all non-commercial users, and finally to everyone, it grew exponentially as LANs were connected. At first many connections used gateways to translate protocols, but eventually common sense prevailed as users switched to TCP/IP and common e-mail protocols. The Internet population would have increased at a much slower rate if users had to first be convinced that connection to a network was useful and then joined one by one.

The growth of the commercial Internet was facilitated by thousands of Internet Service Providers (ISP), often mom and pop companies, that allowed individual users to link to the Internet using dial-up modems. Typically an individual user would be assigned an IP number only when using the service, thus a small number of numbers could be recycled by the ISP. In cities there were typically several ISPs in your area code, while in less populated regions the ISPs established Points of Presence (POP) in multiple area codes so the individual user did not have to pay per minute long distance rates. The decentralized nature of Internet addressing made it easy to add new ISPs to the system.

### Applications for the Internet

telnet

ARPANET was developed for a specific purpose; to enable mainframe computers to be shared among ARPA investigators at different locations. The network of IMPs

and associated network software allowed bits in packets to be sent from one IMP to another, but you need application software to actually accomplish a task. After the network was up and running, it was natural that the first application to be written was **telnet** (telecommunications network), which emulated a terminal connected directly to the mainframe. In 1970 telnet was being used to run jobs remotely over ARPANET, and by definition was the first Internet killer app.

While each IMP and the network software it uses is the same, the telnet application is running on and is thus specific to the model of mainframe; thus a different telnet must be written for each type of machine. However, the telnet application is fairly simple, since it assumes a direct, reliable link to the other computer, it doesn't need to know anything about networks or packets; the two computers could be in adjacent rooms and linked by short wires as far as it is concerned. The IMP network was set up for just this purpose; to make the actual applications simple.

### ftp

An application to transfer files between computers, **ftp** (file transfer protocol), was written in 1972. The files could be data, e.g. input or output for computer programs, or the computer programs themselves. This made collaborations between users at different computer sites much easier. The engineers developing the Internet found that ftp was very useful for exchanging messages and the RFC documents that summarized their work. This was the beginning of e-mail. Network supervisors discovered to their surprise that e-mail soon became a major portion of data exchanged over ARPANET, and thus the second killer app.

### e-mail

However, ftp was only the beginning of e-mail. In 1982 **smtp** (simple mail transfer protocol) was developed. It incorporated the now familiar e-mail addresses for individuals and notified the sender if mail could not be delivered. Subsequent applications enabled workstations and PCs, which were only connected to a network for a few hours a day, to retrieve mail from a **pop** (post office protocol) server when they connected. The MIME (Multipurpose Internet Mail Extension) was developed in 1996 to carry the characters used by European, middle Eastern and Asian languages, as well as complex formats that contain images. The enhancements just made e-mail a bigger killer application.

### World Wide Web

This brings us to the third Internet killer app, the World Wide Web (**www**). In 1989, Tim Berners-Lee, a physics major who had been writing commercial software for several years, but was then working at CERN, the high energy accelerator center at Geneva, Switzerland, proposed a system for exchanging information among scientists and engineers. The client, called a browser, would request a page of information (the page could be smaller or larger than the amount that could fit on the screen), and the server would send that page. The page was displayed using a markup language that included links; if you used a mouse to click on a link, another page would be

requested and then displayed. The application required a GUI by its nature, and was developed on a NEXT computer (a Unix based machine created by the company started by Steve Jobs after he was dismissed from Apple Computer). Berners-Lee credits the elegant (at least for the time) GUI and the application development environment of the NEXT for his success. The www was a powerful system for collaborations between multiple authors writing scientific papers or engineering documents, since comments and alternate versions of sections could be incorporated into a main document. A working version of the web was on the CERN network in December of 1990.

The World Wide Web, as seen through its browser, attracted immediate attention among the users of the Internet, at that time scientists and engineers using Unix machines. Two important enthusiasts were Marc Andreessen and Eric Bina, undergraduates at the University of Illinois with jobs at the National Center for Supercomputer Applications. They wrote an improved version of the browser, which they called Mosaic. Perhaps the most obvious new feature was the ability to generate pages containing mixed text and images, just like a book or magazine (in the original browser images were displayed only as separate pages). By 1993 versions of Mosaic had been written for MS Windows and Macintosh computers, thus greatly expanding the population that could browse the World Wide Web.

#### The commercial www

The general public was becoming aware of the Internet as an ideal medium for carrying e-mail, but the emergence of the www and a browser that displayed illustrated pages generated a new burst of excitement. Andreessen moved to California and was contacted by Jim Clark, the computer scientist and entrepreneur who founded Silicon Graphics. Clark realized that the www would be immensely popular, and he had money and connections in Silicon Valley. The result was an instant company, Netscape, which hired programmers and created the first commercial browser, Netscape Navigator. The browser went on sale for \$39 in October, 1994, and Netscape had a wildly successful Initial Public Offering in 1995.

By this time the people at Microsoft had become aware that a sea change had occurred in the personal computing environment, and they felt left out. A www browser could be thought of as a GUI which displays and accepts information transmitted in an open, standard format. In fact, manufacturers of instruments, routers, storage devices, etc. began to use http, html and a web browser for communication with computers. This freed them from the need to write multiple control and display applications using the APIs for each computer platform. This www approach was independent of the operating system, and anything that is independent of MS Windows does not set well with Microsoft. It may not have been diplomatic for Andreessen to announce that MS Windows would soon be reduced to a collection of device drivers for the www browser.

Microsoft responded to the challenge in a characteristic manner. They acquired Spyglass, a company that had written a web browser, pumped in money and people and produced its own www browser, Internet Explorer, a www browser with versions for MS Windows, the Macintosh, etc. Despite the considerable expense of producing

Internet Explorer, it was given away free! To drive the nail in the heart of Netscape, all computers sold with the MS Windows operating system had Internet Explorer preinstalled, and it was difficult to use any other browser on the system. Microsoft was sued by the US Government for monopoly activities, and in a protracted litigation Microsoft claimed that its www browser was such an integral part of its operating system that it would be impossible to remove or replace it and still have a functional operating system. The Republicans gained control of both the executive and legislative branches of government in the year 2000, and antitrust litigation against Microsoft trailed to a finish.

Netscape did not fare well. For a while they switched emphasis to enterprise software, then their downward spiral was halted, or at least hidden, by acquisition by America On Line (AOL). Netscape Navigator was now offered free of charge; few people would pay for a browser when Internet Explorer was free. In addition, Netscape Navigator acquired considerable clutter as it became loaded with advertisements supplied by its parent corporation, which changed from AOL to AOL Time Warner. It played a minor role in the www community, after dilution by the Internet Explorers loaded onto every new PC.

Why is Internet Explorer free?

The Microsoft anti-trust case was complicated; there were many technical and legal issues. As would be expected, the press, which followed the litigation closely, emphasized personal issues; the apparent inconsistent and incompetent Microsoft defense and the inarticulate testimony (by video) of Bill Gates, the Chairman and founder of Microsoft. But here we want to focus on one issue, one that was scarcely mentioned by the Press: why would Microsoft force Internet Explorer on users? Maybe Microsoft wanted all users of MS Windows to have access to the www even if they couldn't afford Netscape Navigator? A more believable suggestion is that Microsoft could charge more for its MS Windows if it also contained a www browser. However, the price of MS Windows didn't increase. The final, and only plausible motive for creating a MS Windows proprietary browser involves the functional definition of www protocols and formats, and is thus an object lesson for this book.

As has been explained previously, the www browser and the www server function as a team. The browser requests information using http protocols and the server sends information using http protocols. The browser then displays the information and responds to user commands, (clicking with the mouse, etc.). If you want to change the protocols, you would write your own browser which used different protocols. Of course, new protocols would make no sense unless you also written your own server application which would respond to these new protocols.

Now we close the circle. In addition to the browser, Microsoft sells a www server application, Information Exchange, for computers that are running MS Windows as their operating system. In contrast to the browser, the www server application makes real money for Microsoft. In addition to the standard www commands, the Internet Explorer browser and the Information Exchange server can use a set of "enhanced" commands that are not part of the standard set. Thus Microsoft has started the evolution of the www into a version that can only be fully used by Microsoft operating

systems and applications. Microsoft has started to create the type of environment maintained by IBM several decades earlier, proprietary standards for networks: the environment that the Internet was created to bypass.

## JAVA

The JAVA language was created by Sun Microsystems in the early 1990s as an error resistant language for embedded processors, i.e. the structure of the language makes it difficult to write code that will crash at run time. After failing to generate interest among manufacturers of devices that might use JAVA, especially TV set top boxes that would implement interactive television (an oxymoron?) Sun decided JAVA had potential for use on the Internet. The feature of JAVA that made it appealing was that it was implemented in a two step process which made it independent of the type of machine it was running on. JAVA was said to allow a programmer to "write once run everywhere".

As a typical high level language, a JAVA program is first written in a syntax, the source code, that is easily understood by human programmers. Most high level languages are then translated by compilers into machine instructions that can be executed by the computer the application runs on. By definition, these machine instructions are specific to a particular type of CPU. In contrast, JAVA source code is translated by its compiler into a standard, symbolic "byte code", which is intended to run on a virtual machine. The virtual machine is actually a computer application that is running on the computer that is running the JAVA code. This strategy is not quite as efficient as generating machine code, since the byte code must be translated by the virtual machine application into machine code. However, it's not as bad as it may seem because the byte code instruction set is similar to a typical CPU instruction set. Thus, the virtual machine only needs to implement a very simple translation, and simple means fast. The bottom line is that JAVA is slower than most compiled languages, but not that much slower.

One of the first dramatic uses of JAVA was to create applets. An applet is a little JAVA application which can be imbedded in a web page just like an image is imbedded in a web page. As long as the users computer has a JAVA virtual machine application, the JAVA application can execute and display its results on the web page. The JAVA applet can generate complicated animation, and the animation can be controlled by the user, e.g. rotation of a 3-D object in response to clicking on arrows. An applet can function as a calculator, for example calculating monthly payments on a mortgage where the user enters the total mortgage and percent interest. While JAVA applets were initially perhaps the most dramatic use of JAVA, the language has since become the backbone of many business users of the Internet.

## Microsoft invents C# and .NET

At first Microsoft included a JAVA virtual machine as part of Internet Explorer. However, in 2001 they announced that new versions of the Microsoft browser would not translate JAVA code, but rather translate the code of a new language, created by Microsoft, C# (pronounced see sharp). The C# language was part of a larger ".NET"

framework, which would allow code written in many languages to be mixed and run over the Internet: as long as you were using a MS Windows machine. The Microsoft philosophy was "any language, one platform": and the platform was and is MS Windows. Associated with the .NET program, all Microsoft applications would not be actually purchased, but rather leased on an annual basis. If you lease the software upgrades are automatically downloaded from the Internet, if and when upgrades are generated. If you don't renew the lease, the software doesn't run. There is no longer an option of not buying upgrades, they are part of the lease.

### Distributed computing

If the Internet can be used for transmitting information between computers, it can be used to create groups of computers that work on the same problem, i.e. are running the same application. A dramatic example of this kind of computing is the SETI (Search for Extra-Terrestrial Intelligence) project. The SETI agency maintains large antennas that receive micro-wave signals from space. In order to identify signals that might come from extra-terrestrial beings, the signals are processed by several mathematical filters, the main one being a Fourier analysis to look for periodicity. This is very computer time intensive; however, there are thousands of PCs connected to the Internet that use their CPUs only a small fraction of the time. Even as I am writing this book, my CPU is mostly idle; it works for less than a millisecond after each of my keystrokes. Users that elect to help in the analysis of SETI data first download an application from the SETI site. Then they periodically download small chunks of data and upload the analysis of this data that was obtained using the SETI application. This operation of this distributed computational machine, SETI@HOME, is essentially transparent to the user. They can use their computer at apparently the normal speed even while the SETI data is being processed.

If SETI@HOME is a distributed computational machine, NAPSTER is a distributed data base. Users of NAPSTER are able to share digital music files (typically in the MP3 format) by downloading the NAPSTER application and then allowing a section of their hard disk containing the music files to be accessed by other NAPSTERs over the Internet. The music files are not transferred to a central site, they remain on the owners machine and are thus transferred between the individual machines. As you can image, the commercial music producers, the CD manufacturers, were not enthusiastic about NAPSTER, and used the legal machinery to shut it down. NAPSTER was replaced by similar networks, but networks that didn't even have a central distribution site for the application; these distributed networks could not thus be shut down, only threatened. The music industry also agreed to distribute music over the Internet, but only for a fee. Many copy protection schemes were proposed, created, and broken. Who knows what will eventually evolve in the music world, but development and use of this type of distributed system is clearly the important, and versions of it can be used for many purposes.

### The future of the Internet

At the time I am writing this, 2004, many of the Internet based companies founded with venture capital in the late 1990s have disappeared. The NASDAQ stock index,

rich in technology stocks, has declined from a high of over 5,000 to less than 1,500. The large number of computers, routers, miles of optical fiber, and computer technicians that were created to support these companies are warehoused or have been recycled. However, the Internet revolution has not been reversed. Most of the people in the US and a large fraction of those in the Industrialized world use the Internet for e-mail. Amazon.com is still growing and now even makes a small profit. Most airline tickets are sold over the Internet. National and local governments are continuing to increase use of the Internet to exchange information with the voters.

It is clear that commercial forces are huge, but their business plan are typically unproved. There is a desire to mold the Internet into a mass consumer machine similar to commercial TV. Microsoft isn't the only, just the strongest, computer company that would like to have a monopoly of users. As long as there is some balance between corporations maybe a complete monopoly can't occur. It seems unlikely, even if the Democratic party were in control in the US, that the government will play an important role in maintaining a balance. However, it seems almost certain (to me anyway) that a free and universal Internet will remain available to those that want and need it the most, e.g. the academic community.



Internet time line. Events associated with development of the Internet are arranged chronologically:

Column 1- ARPANET

- 1957 Russian satellite Sputnik motivates defense technology spending
- 1958 ARPA (Advanced Research Program Agency) dispenses money
- 1960 Paul Baran proposes bomb proof networks that distribute small packets
- 1967 first funds for ARPANET to allow remote users to share mainframes
- 1969 BBN wins ARPA contract to program IMP minicomputers;
- 1969 First 4 ARPANET nodes: UCLA, SRI, UC Santa Barbara and U of Utah
- 1970 telnet application allows remote users to run programs
- 1972 ftp (file transfer protocol) application; start of e-mail;
- 1972 ICCC demonstration of applications running on ARPANET
- 1975 TCP implemented
- 1978 IP split from TCP; now a separate layer
- 1983 ARPANET switches from NCP to TCP/IP
- 1985 domain servers replace the NIC as source of IP addresses
- 1990 ARPANET is closed, merges with the Internet

Column 2- non-ARPANETs

- 1965 Intelstat, first US geostationary communication satellite
- 1970 AlohaNet, packet radio network linking Hawaiian computer centers
- 1972 Cyclades, French network using Internet-like protocols
- 1975 PRNET, (Packet Radio Network), SATNET, (Satellite Network), and ARPANET form first Internet
- 1981 CSNET (Computer Science Network) links non-military computers
- 1983 MILNET (Military Network) takes over military traffic from ARPANET
- 1985 NSFNET (National Science Foundation Network); an extension of CSNET
- 1990 first www (world wide web) started at CERN (Geneva)
- 1991 CIX (Commercial Internet Exchange) formed: a commercial NSFNET
- 1993 Mosaic is the first www browser for MS Windows, Mac, etc.
- 1995 all Internet traffic now carried on CIX, no limitation on content
- 1995 Amazon.com sells books; the first major retail sales site
- 1999 SETI@Home: first distributed computing network
- 1999 Napster: first distributed file exchange network

Column 3- hobby nets, information providers, and big business

- 1979 Compuserve provides information and e-mail
- 1979 first private computer bulletin board
- 1983 FidoNet links bulletin boards by telephone
- 1985 The Well, the hip bulletin board
- 1989 AOL (American On Line) competes with Compuserve with graphics
- 1998 AOL buys Compuserve, the strong buy the weak
- 2001 AOL merges with Time Warner

Column 4- the hardware

MAINFRAMES

- 1946 ENIAC I, used by military, the first programmable computer
- 1951 UNIVAC, the first civilian computer, used in census
- 1959 IBM 7090, first computer using transistors

1964 IBM 360, the first computer family

### MINI-COMPUTERS

1965 PDP-8 (Digital Equipment Corp)

1969 H-516 (Honeywell) used as the IMPs for ARPANET

### PERSONAL COMPUTERS

1975 Altair 8800 (kit) first PC

1976 Apple II, the first mass produced PC

1981 IBM PC, the name and a professional display

1982 Sun engineering workstations; network capable and UNIX

1983 Ethernet, the most popular network software/hardware

1984 Apple Macintosh, the first PC with a GUI and networking