



Welcome back to the third in a series of articles reviewing the history of personal computing. In the summer issue of *CHIPS*, we looked at the development of the modern personal computer. In the fall issue, we examined the evolution of personal computer (PC) operating systems and application software. In this issue, we will look at the technologies that tie our PCs together through networking.

We tend to think of digital networking as a relatively new concept, but the roots of modern networking extend back over 150 years. Many years before Charles Babbage created what is considered the first computer, his "Differential Engine," the telegraph ushered in the age of digital communications in 1844 when Samuel Morse sent a message 37 miles from Washington D.C. to Baltimore using his new invention. While the telegraph is a long way from today's computer networks, it was arguably the single most significant event in human communication since the development of language. For the first time in human history, we had a reliable method of communicating in real-time beyond line-of-sight. As long as you could connect two locations with wires, you could exchange information almost instantaneously without regard to distance. Much as modern data networks use 1s and 0s to encode and transfer information, Morse code was the language of the telegraph. Morse code is a binary-like system that uses dots and dashes in different combinations to represent letters and numbers. The big difference is, that while the telegraph operators of the mid-19th century could perhaps transmit four or five dots and dashes per second, computers now communicate at speeds of up to one billion 1s and 0s every second, which we refer to in digital shorthand as one gigabit or "1Gb."

Not long after Morse invented the telegraph, a Frenchman named Emile Baudot developed a typewriter-style telegraph machine that allowed users to key in their messages using the basic alphabet and print out received messages using automatic translators built into the machine. These early precursors to modern modems allowed virtually anyone to send and receive telegraph messages without having to understand the code used to transmit the message. However, Morse code did not lend itself well to automation due to the variable length of each character, so Baudot developed a more uniform code for his system. Baudot used a five-bit binary code to represent each character. As that only gave 32 possible characters (00000 to 11111 = 32), it wasn't

going to be enough to include all 26 letters and 10 digits. He solved this problem by adding two "shift characters" for figures and letters that performed in much the same way as a typewriter shift key. This gave him 62 combinations (not quite six-bit computing) for letters, figures and punctuation marks. Western Union, the most famous telegraph company in history, eventually replaced all of its Morse telegraph equipment with Baudot's "teletypewriters." In honor of Baudot's pioneering contributions, the speed of serial communications is still measured today by measuring the "Baud rate."

However, despite being the dominant digital communications code for over a century, the Baudot five-bit code was not suited to 20th century computing. Computers, which were developed independently of the telegraph, needed the ability to discriminate between upper and lowercase letters. Baudot's code only provided for uppercase letters. In response to the need for a new standard information exchange format, a group of American communications companies got together in the 1960s to devise a new code. Their new standard used seven bits that could represent 128 characters. This new standard came to be known as the American Standard Code for Information Interchange (ASCII). ASCII was immediately accepted by virtually everyone in the communications world, with one notable exception: IBM. IBM decided to make its own standard, the Extended Binary Coded Decimal Interchange Code (EBCDIC). The IBM code used eight bits and could represent 256 characters. However, aside from IBM using it in their mid-range and mainframe computers, EBCDIC never really caught on. Once it became clear that IBM would not be able to force their proprietary standard on the rest of the world, they eventually adopted the ASCII code. However, as they still wanted the extra capabilities inherent in the 8-bit format, they "extended" ASCII by using an eighth bit so it could represent 256 characters and called it "Extended ASCII." Now that a common language for computer data had been invented, the stage was set for real computer networking to begin.

## Early Networking

The origins of the Internet were distilled from the visions and work of computer visionaries of the 1960s. Three of the most influential were the Massachusetts Institute of Technology (MIT) trio of J.C.R. Licklider, Leonard Kleinrock and Lawrence Roberts. Licklider first proposed a global network of computers in 1962. Later that year he moved to the Advanced Research Projects Agency (ARPA) to head the work to develop it. Kleinrock developed the theory of packet switching, which would form the basis of Internet connections. Roberts confirmed Kleinrock's theory in 1965 when he connected a Massachusetts computer with a California computer over dial-up telephone lines. However, while this demonstrated the feasibility of wide area networking, it also showed that the circuit switching technology available through a standard telephone line was not sufficient to support any large-scale networking. Shortly after this project, in 1966, Roberts began work at ARPA and developed the plan for what eventually became ARPANET.



## Finding True Believers

When ARPA sent out a request for proposals to build the initial network of four Interface Message Processors (IMPs), many of the large computer and telecommunications organizations did not bother responding, because they thought the task was impossible. Turning ARPA's networking theory into reality fell to another group of visionaries at a small company named BBN (Bolt, Beranek and Newman). We take much of the support activities that sustain the Internet for granted today, but BBN literally created most of them from scratch. They wrote code that would automatically reload crashed servers, pull packets into the machine, figure out how to route them, and send them on their way. They also developed a routing scheme that would automatically route data packets around troubled links in the network and update itself several times per second. BBN had to handle some stiff challenges, not the least of which was dealing with the timing and error-control problems associated with sending data over telephone lines. This was pretty cosmic stuff in an era where most engineers still carried a slide rule and the microprocessors that power modern CPUs had not been invented yet.

The key to the design of ARPANET was the construction of an autonomous subnet, independent of the operation of any host computer. An IMP can take on one of two distinct roles: Host or "store-and-forward." In any host-to-host connection, the IMPs at the respective host sites are the source and destination IMPs for that connection, and the IMPs in the network path between the host sites comprise the store-and-forward sub-network. The IMPs of the sub-network received packets, performed error control, determined the route and forwarded them to the next IMP in the network path. In addition to these tasks, the source IMP and destination IMP were responsible for end-to-end connection management and message processing procedures for the duration of the connection. These procedures included flow control, storage allocation, and message fragmentation and reassembly.

There were many factors that affected the development of message processing requirements. First, there was some likelihood of delay in acknowledging packets due to finite bandwidth or differing bandwidth at the source or destination. This would result in packets arriving out of order, becoming duplicated if they weren't acknowledged by the receiving host in time, or becoming just plain lost. Also, IMPs only had a limited amount of storage space, so they needed to pass packets on as quickly as possible. After spending months customizing software and systems, BBN eventually got the first two IMPs set up at the University of California at Los Angeles and Stanford. ARPANET was born on October 1, 1969, when the first characters were transmitted over the new network. The network quietly expanded to 13 sites by January 1971 and 23 by April 1972.

Outside of BBN and a small group of researchers, ARPA, the network that would transform the world was virtually unknown until the International Conference on Computer Communication in Washington, D.C., October 1972. The ARPANET was the only dem-

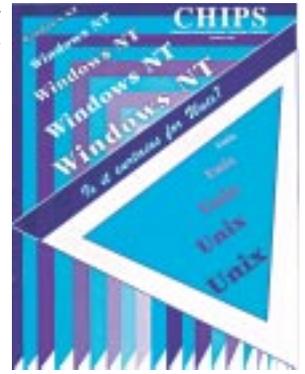


onstration at the conference and conclusively proved the feasibility of packet switching networks. Though most of the world still did not know it, we had taken our first steps toward wiring the world for data.

## Ethernet

The next big development in networking after ARPANET and packet switching was Ethernet, which is still the dominant network technology today. The roots of the modern Ethernet were planted in a 1973 Xerox Corporation patent memo that described a new protocol for multiple computers communicating over a single cable. Originally intended to help design internal computer-to-computer communications within Xerox copiers and duplicators, Ethernet eventually became a global standard for interconnecting computers on local area networks. Ethernet was developed by Xerox at their Palo Alto Research Center (PARC) in California. In 1979, Digital Equipment Corporation and Intel joined forces with Xerox to standardize the Ethernet system. The first specification by the three companies, called the "Ethernet Blue Book" was released in 1980. Ethernet was originally a 10 megabit per second system (10Mbps = 10 million 1s and 0s per second). It used a large coaxial backbone cable running throughout the building, with smaller coax cables attached at short intervals (usually around six feet) to connect to the workstations. The large coax became known as "Thick Ethernet" or "10Base5." The "10" refers to the speed, which in this case is 10Mbps. "Base" means it is a base band system that uses all of its bandwidth for each transmission, as opposed to a broadband system that splits the bandwidth into separate channels to be used concurrently. The "5" refers to the system's maximum cable length, in this case 500 meters. In 1983, the Institute of Electrical and Electronic Engineers (IEEE) released the official Ethernet standard, IEEE 802.3. This second version is commonly known as Thin Ethernet or 10Base2 (10Mbps, base band, 200 meters).

In 1985, the Computer Communications Industry Association (CCIA) asked the Electronics Industries Association (EIA) to develop a cabling standard which would define a generic telecommunications wiring system for commercial buildings to support a multi-product, multi-vendor environment. This would be a cabling system that would run all current and future networking systems over a common topology using a common media and common connectors. By 1987 several manufacturers had developed Ethernet equipment that could utilize twisted-pair cable, and in 1990 the IEEE released the 802.3i Ethernet standard 10BaseT (T refers to twisted-pair cable). In 1991 the EIA together with the Telecommunications Industry Association (TIA) published a standard for telecommunications cabling (EIA/TIA 568). It was based on Category 3 Unshielded Twisted Pair cable (UTP), and was closely followed one month later by a Technical Systems Bulletin (TSB-36) which specified higher grades of UTP cable, Cat 4 and Cat 5. Cat 4 specified data rates of up to 20MHz and Cat 5 up to 100MHz, which at the time seemed like a lot of bandwidth. However, as George Carlin observed, "stuff accumulates to fill available space." Given the exponential growth of networking tech-



nology, even Cat 5 is being pushed to its limits. The current state of the art is Cat 6, and Cat 7 is waiting in the wings.

Despite being pronounced “about to be dead” several times in the last 15 years, Ethernet has successfully defended itself against all comers in the networking standards world, including LAN Token-Ring, Fiber Distributed Data Interface (FDDI) and Asynchronous Transfer Mode (ATM). You can tell who is winning simply by looking at the type of equipment people are buying. Network interface cards (NICs) and switches are generally replaced every two to three years. Since 1998, 90 percent of all NICs and switch ports shipped have been some flavor of Ethernet. Case closed, at least for now.

There are two basic reasons Ethernet still rules. First, the invention and installation of fiber-optic cable, with its huge bandwidth potential, means you can use a “cheaper, dumber” technology like Ethernet as efficiently as “expensive, smart” technology like ATM. Without fiber optics, we would need all of the ATM horsepower to squeeze every last drop of data into the scarce bandwidth available on copper wire. With fiber, that bandwidth constraint has pretty much gone away. Also, Ethernet has been getting smarter in useful ways. Because Ethernet adapters can auto-sense 10Mbps, 100Mbps and 1,000Mbps operations, it’s now possible to establish a tiered Ethernet network that supports all three speeds using the same standard. For example, a LAN may have a Gigabit Ethernet backbone and departmental servers that are connected by Fast Ethernet, and then connected to conventional 10Mbps Ethernet switches and hubs that tie into desktops. Without that ability to automatically sense what speed your backbone is using, we might need to integrate three different network protocols to do the same thing Ethernet does on its own. There are other technologies and standards that I really wish we had time to review here, including FTP (file transfer protocol) and TCP/IP (Telecommunications Protocol/Internet Protocol, also developed at BBN). But the issue I’ve saved for last that incorporates both of those issues is the Big Kahuna of networking — e-mail.

## You’ve Got Mail!

Seventeen years ago, when I first started fooling around with computers, the only people who had e-mail were the few thousand hardy souls who had access to ARPANET or large private or corporate systems like General Electric. Everything was in plain text and files were exchanged via FTP over Unix-based systems. Think about this: the Internet, with its millions of servers, and the World Wide Web, with its billions of pages, are all essentially the result of the human desire to communicate. I submit to the jury that e-mail, more so than any other single factor, is the application that is primarily responsible for the development of the modern Internet. Here is my case. E-mail first appeared in the 1960s when users on time-sharing systems wanted a way to leave messages for each other. These early e-mail systems were very simple. Mailboxes consisted of a text file, readable only by a single user, to which new messages were appended. There were no mail reader programs. Users had to scroll through the text file to the most current entries. If the reader didn’t edit out old mate-



rial fairly frequently large mail files could become very long and hard to get through. These primordial e-mail systems were initially limited to the physical reach of the local system.

ARPANET added “reach” to e-mail by connecting systems together. The first recorded case of e-mail traveling from one site to another occurred in 1972 when Ray Tomlinson, then an engineer at BBN, delivered an electronic message by copying it across a network link connecting two DEC PDP-10s. Tomlinson, by the way, is also the person who decided to use the “@” symbol to separate the user from the host part of an e-mail address. E-mail caught on quickly. Less than a year later, 75 percent of the traffic on the ARPANET was e-mail. There were no protocols that specifically covered e-mail. Mail was sent via FTP, which had commands specific to mail transfer. Mail delivery and tracking information was included in the mail headers, but no defined mail header standards. Also, mail programs that disagreed over formats would often refuse to talk to one another. For example, Multics systems used the @ symbol as a “line kill” command.

At the time of most of these events, TCP/IP, which eventually provided a standard exchange format for all networks, had not yet appeared on the scene. The ARPANET used Network Control Protocol (NCP) as its core network protocol, and was not able to communicate with any other packet network in existence at the time. Deliverance from the e-mail Tower of Babel first appeared in the form of “delivermail,” which was developed by Eric Allman and originally shipped with BSD (Berkeley Software Distribution) Unix versions 4.0 and 4.1 in 1979. Delivermail successfully handled e-mail using FTP over NCP and was soon incorporated into the ARPANET community. Delivermail eventually evolved into sendmail, which is arguably the most influential and important e-mail program developed to date.

About the same time that e-mail was developing on ARPANET, Vint Cerf (Father of the World Wide Web) and Bob Kahn (from BBN) were working on a way to connect packet networks together. The results of their work would become the TCP/IP protocol, which defined standards for data exchange and communication between networks. ARPANET transitioned to TCP/IP in 1982, and the widespread implementation of TCP/IP paved the way for today’s standard for e-mail: Simple Mail Transfer Protocol (SMTP). In response to the development of SMTP, Allman evolved his delivermail program into sendmail, which extended the reach of e-mail beyond the ARPANET system and allowed users to communicate between all the various private packet networks that would eventually form what we now know as the Internet. The drive to communicate, coupled with the development of a universal system of point-to-point communications embodied in e-mail, are what brought the Internet together.

Billions, perhaps trillions of dollars have been spent over the last 150 years devising faster, more robust ways to allow people to set lunch dates, ask “whassup,” send sales pitches, and — squabble. E-mail further evolved in the 1990s with the introduction of more feature-laden mail programs, including Lotus ccMail and Notes, Microsoft Outlook, and various other programs. But other than adding the ability to transmit richer types of information (including, unfortunately, potentially hostile payloads), they have basically just extended the functionality originally codified by sendmail and SMTP. The final evidence in support of my belief in

e-mail's pivotal role in the development of modern networking is this: current estimates from people who watch Internet traffic patterns say that the Internet will pass over 36 billion e-mails this year. That comes out to 114 e-mails to roughly a second, every second of the year. And that figure will only grow as more areas of the world gain access.

## Closing Words

There are various opinions on what it takes to build a network, but one that caught my eye recently was offered by Van Macatee, an executive at Level 3 Communications, in the November 1, 2002, issue of the Web magazine *America's Network*: "*Any schmuck can build a network.*" I'm not sure how Macatee defines a schmuck, so I'll offer a definition: a network schmuck is someone who knows what the technology can do and how to plug it in and turn it on, but not how the technology works or what effect it will have on the people connected to it.

The Internet, is only relatively simple today because of the efforts of the pioneers in the field who had the vision to see the future, the skills and will to make it happen, and the wisdom to cooperate to achieve common goals. The development of the hardware, software, and transport protocols and technologies that make up modern networking are the products of many dedicated, intelligent, talented people whose efforts rival the building of the Pyramids and the Apollo space program as cooperative human endeavors. Schmucks did not build the Internet. Despite the probable difference in our salaries, I strongly disagree with Macatee's assertion. Perhaps just about anyone can buy a network out of a box and just plug it in. But plugging in and turning on a network are not the same as building one.

A modern parallel to the development of the Internet is the Navy's NMCI project. The goal is similar: build a single extended network to serve the entire service in much the same way that the Internet now serves the world. The Navy has many of the same challenges in building the NMCI that faced the people who built the Internet: defining common standards, integrating technologies, and getting everyone to agree on the one right way to do certain things. The Navy is at a pivotal point. In building the NMCI you can, right now, shape the work environment of the entire Navy for decades to come. Please remember, though, that simply building a big network that adheres to a single set of technical standards is not the goal. NMCI will ultimately be judged on how it supports the Navy as an organization. What the world has done with the Internet, I believe can be done with NMCI.

That's all for now. In the next issue, we will conclude this serial history of personal computing with a look at the development of the World Wide Web and what it means to be part of today's wired, interconnected world. Until then...



## Happy Networking!

*Long is a retired Air Force communications officer who has written for CHIPS since 1993. He holds a Master of Science degree in Information Resource Management from the Air Force Institute of Technology. He is the Telecommunications Manager for the Eastern Region of the U.S. Immigration & Naturalization Service.* □

## Talking with Dinah Cohen Computer/Electronic Accommodations Program (CAP) Director

CHIPS: Many people talk about the "digital divide" separating those with access to computers and the Internet to those who do not have access opportunities for financial reasons. But isn't there another digital divide separating private citizens with disabilities from technology? CAP does such a great job assisting DoD and Federal employees with disabilities to bridge the gap, but is there an agency to assist private citizens with disabilities who may be cut off from technology?

Ms. Cohen: The digital divide falls into two categories. The first, people who have access to a computer, but cannot access the information. I hope and think that Section 508 is reducing this divide by working with industry and Federal Government to ensure that electronic and information technology is accessible and usable by people with disabilities. For assisting people in obtaining access to a computer, I am aware of some bold actions regarding universal design and assistive technology/computer access that are part of President Bush's New Freedom Initiative. You can see more on this issue at [www.disabilityinfo.gov](http://www.disabilityinfo.gov).

Editor's Note: The New Freedom Initiative was established to ensure that the more than 54 million Americans with disabilities learn and develop skills, find meaningful work, and realize the promises of the Americans with Disabilities Act. To achieve equality of opportunity, independent living, and economic self-sufficiency, this comprehensive plan promotes the full participation of people with disabilities in all aspects of American life. The Federal Web site, [www.disabilityinfo.gov](http://www.disabilityinfo.gov), provides resource information and links to agencies and programs designed to assist citizens with disabilities. Just a sampling of links follow. Comprehensive information about Federal job opportunities can be found at [www.usajobs.opm.gov](http://www.usajobs.opm.gov) or call 1-478-757-3000/TDD 1-478-744-2299. A free service of the Office of Disability Employment Policy (ODEP) of the U.S. Department of Labor, the Job Accommodation Network available at [www.jan.wvu.edu](http://www.jan.wvu.edu) or 1-800-526-7234 (V/TTY), provides information about job accommodations, the Americans with Disabilities Act (ADA), and the employability of people with disabilities. The RESNA Alternative Financing Technical Assistance Project (Agreement No. H224C000200) is funded by the National Institute on Disability and Rehabilitation Research (NIDRR) under Title III of the Assistive Technology Act of 1998. This Web site, [www.resna.org/AFTAP/index.html](http://www.resna.org/AFTAP/index.html), was developed with grant funds and is designed to assist individuals in receiving loans to ensure they can access assistive technology. The information on these pages does not necessarily reflect the position of NIDRR/U.S. Department of Education or RESNA, and no official endorsement of the materials should be inferred. □