CREATING A MATHEMATICAL THEORY OF COMPUTER NETWORKS

LEONARD KLEINROCK

University of California at Los Angeles, Computer Science Department, 405 Hilgard Avenue, Los Angeles, California 90095-1361, lk@cs.ucla.edu

1. ORIGINS

It all began with a comic book! At the age of 6, I was reading a Superman comic at my apartment in Manhattan when, in the centerfold, I found plans for building a crystal radio. To do so, I needed my father's used razor blade, a piece of pencil lead, an empty toilet paper roll, and some wire, all of which I had no trouble obtaining. In addition, I needed an earphone, which I promptly appropriated from a public telephone booth. The one remaining part was something called a "variable capacitor." For this, I convinced my mother to take me on the subway down to Canal Street, the center for radio electronics. Upon arrival to one of the shops, I boldly walked up to the clerk and proudly asked to purchase a variable capacitor, whereupon the clerk replied with, "what size do you want?" This blew my cover, and I confessed that I not only had no idea what size, but I also had no idea what the part was for in the first place. After explaining why I wanted one, the clerk sold me just what I needed. I built the crystal radio and was totally hooked when "free" music came through the earphones—no batteries, no power, all free! An engineer was born, and the seeds for the Internet technology were sown.

I spent the next few years cannibalizing discarded radios as I sharpened my electronics skills. I went to the legendary Bronx High School of Science and appended my studies with courses in radio engineering. When the time came to go to college, I found I could not afford to attend, even at the tuition-free City College of New York (CCNY), so I enrolled in their evening session program in electrical engineering while working full time as an electronics technician/engineer and bringing a solid paycheck home to my parents. My work and college training were invaluable and led to my winning a full graduate fellowship to attend the Massachusetts Institute of Technology in the Electrical Engineering Department.

2. THE MIT ENVIRONMENT

At MIT, I found that the vast majority of my classmates were doing their Ph.D. research in the overpopulated area of information theory. This was not for me, and instead I chose to break new ground in the virtually unknown area of data networks. I chose this area because I was surrounded by computers at MIT, and it was clear to me that some technological breakthroughs were necessary to allow them to communicate with each other efficiently. In 1961, I submitted a Ph.D. proposal (Kleinrock 1961a) to study data networks, thus launching the technology that eventually led to the Internet. In the middle of 1961 I published the basic paper (Kleinrock 1961b) laying out the beginnings of the mathematical theory of data networking, introduced the ideas of segmenting messages into smaller pieces (later called "packets") in early 1962 (Kleinrock 1962a), and completed my Ph.D. work in 1962 (Kleinrock 1962b), which was later published in 1964 by McGraw-Hill as an MIT book entitled *Communication Nets* (Kleinrock 1964). In these works, I developed the theory of stochastic flow of message traffic in connected networks of communication centers and developed the basic principles of packet switching, thus providing the fundamental underpinnings for the Internet technology. When I use the phrase "Internet technology," I intend it to refer to the fundamental analytic and design principles and algorithms, and not to a wider use of the term which might include, for example, the World Wide Web, HTML, Java, etc.

I set up the mathematical model using queueing theory, introduced the critical Independence Assumption, evaluated network performance, and developed optimal design procedures for determining the capacity assignment, the topology, the routing procedure, and the message size. I introduced and evaluated distributed adaptive routing control procedures, evaluated different queueing disciplines for handling traffic in the nodes (specifically, chopping messages into smaller segments, now known as packets), and tested the theory against extensive simulations. The principles I uncovered (along with my subsequent research) continue to provide a basis for today's Internet technology. For this work, I am considered to be the inventor of the Internet technology and one of the fathers of the Internet.

3. THE NATURE OF DATA COMMUNICATIONS

Back in the late 1950s, it was not clear to most engineers and practitioners that data communications were fundamentally different from voice communications. Not only were

Subject classification: Professional: comments on.
Area of review: Anniversary Issue (Special).

0030-364X/02/5001-0125 $05.00
1526-5463 electronic ISSN

these differences unrecognized by most, but also in those cases where they were recognized, the conclusion was that it did not matter because data transmission was of no interest relative to voice transmission. Today, the differences are understood, important, and of such significance that current packet switching networks are seriously threatening the business viability of the 100-year-old voice communication telephone networks.

Voice communication uses a technology called "circuit switching," which requires that a dedicated path of network resources be set up and allocated between the two communicating parties. These resources are dedicated to this voice communication during the entire "call," even if nothing is being spoken by either party. With speech, there is silence on the line approximately 1/3 of the time, and this inefficiency has always been tolerable. However, the nature of data communications is considerably different. Data is inherently "bursty" in that it occurs in short bursts of communications followed by long periods of silence; the ratio of silence to communication can be as high as 1,000:1 or even 10,000:1, and this inefficiency in the use of networking resources is totally intolerable. Indeed, one can characterize data communication users who wish network resources to send their data as follows:

a. they don't warn you exactly when they will demand access

b. you cannot predict how much they will demand

c. most of the time they do not need access

d. when they ask for it, they want immediate access.

It is not surprising that bursty traffic is nasty to deal with. It was clear to me back in the late 1950s that the inefficiencies of circuit switching in handling bursty data traffic could not be tolerated. A new technology had to be invented.

4. THE NEED FOR DEMAND ACCESS

The technology I set out to develop had at its foundation the principle that a user should be assigned a resource (e.g., a communications channel) only when he needs it (i.e., when he actually has data to send). I referred to this as "dynamic resource sharing" or "demand access." Examples of demand access schemes that have been developed are polling, message switching, packet switching, asynchronous TDMA, and CSMA/CD (Kleinrock 1976a). No one had previously elucidated the principles underlying such structures. Moreover, no one had produced a model, much less an analysis, of how they performed under stochastic loads. Lastly, there existed no optimal design procedures for laying out the topology, choosing the channel capacity, and selecting the routing procedure and routes. I did all of the above for the case of demand access to network resources.

5. THE CHOICE OF QUEUEING THEORY

It was clear that message traffic was stochastic and so the tools of stochastic processes would be needed for analysis. But more importantly, I had to develop a mathematical model that reflected this concept of demand access. The basic structure I chose was that of a queue (Kleinrock 1975). A queue is a perfect resource sharing mechanism. It is dynamic, adaptive, and efficient. The server does not wait around for a customer who is not there, but rather provides service to whoever is there needing service. In the case of data communications, the server consists of the resources of the data network (e.g., the communication channels and the switches or routers), the user is the data message or packet stream, and the service rendered is transmission of the message across the data network. Moreover, the quantities that one considers in queueing theory are throughput, response time, efficiency, loss, priorities, etc., and these are just the quantities of interest in data networks. Indeed, it was clear to me that a queue was just the right structure for implementing demand access and that queueing theory was perfect for describing and analyzing data networks.

A. K. Erlang, the father of queueing theory (Brockmeyer et al. 1948), used that theory for representing the behavior of telephone traffic and telephone exchanges (Syski 1960). When operations research appeared and then grew in World War II, queueing theory began to be used in other applications; but telephony was still the dominant application, and most of that work utilized models that consisted of a single queue. What I needed was to consider networks of queues. In the late 1950s, the published literature contained almost no work on networks of queues. Tandem queues (Hunt 1957) had been studied to some extent, as had parallel queues (Morse 1958), but these were not rich enough topologies for data networks. However, a singular exception to this was the work by James Jackson who published a classic paper (Jackson 1957) on open networks of queues. Jackson modeled a "job shop" where the nodes were workstations and the customers were the jobs. He assumed Poisson job input traffic, independent exponential job service times at each of the stations, and paths through the network that were governed by independent transition probabilities among the workstations. He solved for the equilibrium joint distribution of the number of jobs in each of the stations and showed some remarkable properties of the solution. As we see below, I was able to take exquisite advantage of Jackson's result for modeling data networks, but not without a serious modification to the model.

6. MODELING DATA NETWORKS

I chose to model a data network as a network of communication channels whose purpose it was to move data messages from their origin to their destination. Each channel was modeled as a server serving a queue of data messages awaiting transmission. The main metric I used for the performance of the network was T, the average time it took for messages to move across the net. Under extremely general conditions, I was able to show that this mean delay is given exactly by the following equation:

$$T = \sum_i \frac{\lambda_i}{\gamma} T_i \qquad (1)$$

where T is the average network delay (in sec.), λ_i is the average traffic on channel i (in messages/sec.), γ is the network throughput (in messages/sec.), and T_i is the average delay (in sec.) in passing through node i (i.e., channel i). This is an extremely general equation. The next step was to find an appropriate expression for T_i. One would imagine that a simple application of Jackson's open queueing network model would provide the solution. Unfortunately, in Jackson's model it was assumed that the service time at each node in the network was an independently chosen random variable, and this is definitely not the case in data networks because messages and packets basically maintain a constant length as they move from node to node, and the service time (i.e., transmission time) at each node is directly proportional to this length; moreover, there is a definite correlation between the message lengths and interarrival times for the message stream. These dependencies not only invalidate the use of Jackson's model, but they also present an extremely difficult queueing network problem, whose equations of motion I set up. (It took more than a decade and a half for an exact solution to appear for the very special case of a two-node tandem network of equal capacity channels fed by Poisson traffic, and the solution was not an explicit equation (Boxma 1979).) I was faced with the prospect of an intractable problem. To get past this point, I introduced what is now considered a classic assumption, namely my *Independence Assumption* that, in its simplest form, states "each time that a message is received at a node within the net, a new length is chosen for this message independently from an exponential distribution." Without this assumption, the problem was intractable; with this assumption, I was able to use Jackson's model directly, and the full solution dropped right out; I could then model T_i as an $M/M/1$ queue, yielding the following well-known expression for T_i:

$$T_i = \frac{1}{\mu C_i - \lambda_i} \qquad (2)$$

where μC_i is the capacity of channel i (in messages/sec.). Having made this assumption, I then set out to confirm its accuracy and found that it was amazingly accurate for networks in which there was even a moderate degree of connectivity. Moreover, once I admitted an approximation, I could use $M/G/1$ models and other delay components in the (now approximate) analysis to good effect.

7. PRIORITY CONSIDERATIONS, TIME SLICING, AND PACKET SWITCHING

In the course of examining data network performance, it became clear to me that it was important to explore the manner in which message delay was affected when one introduced a priority queueing discipline. I chose to understand this influence in the case of a single node first and then applied the results to the general network case. In this work, I established the first conservation law for queues in which I showed that the sum of the load-weighted mean waiting times for any work-conserving queueing discipline was a constant; this allowed one to understand the delay tradeoff among priority groups in a wide class of queueing systems.

I also showed that the network response time can be improved with packetization, a concept that emerged when I showed how the response time was affected by the length of the message unit; note that this focuses only on response time, and has little to do with efficiency (whereas demand access resource sharing focuses on efficiency). I addressed this issue by showing that round-robin time slicing (essentially breaking a message into smaller messages, later to be called packets) "...results in shorter waiting times for short messages and longer waiting times for long messages ...". Note that the word packet was not coined until later in the 1960s. (It was Donald Davies (1973) from the United Kingdom who coined it while working at the National Physical Laboratories in Teddington, England.) However, packetization by itself does not lead to the underlying technology that supported the Internet. Packetization helps and is part of today's networking technology, but by itself it is not the whole story of the efficiency of networks; rather, the more fundamental gain comes from the introduction of dynamic resource sharing. I must emphasize that the totality of understanding the full picture, and not just the issue of packetization, had to be developed before a convincing body of knowledge could be amassed to prove the case for data networks (Roberts 1999).

8. DESIGN CONSIDERATIONS

Once I had an analytical model for data networks, the next step was to address the issue of optimal design of these networks. The design variables I focused on were choice of capacity for each channel, choice of routing procedure, and topological design. I posed the optimization problem as follows:

$$\text{Minimize} \qquad T = \sum_i \frac{\lambda_i}{\gamma} T_i$$

with respect to:

Channel Capacity Assignment
Routing Procedure
Topology

$$\text{subject to:} \qquad D = \sum_i d_i(C_i)$$

where

$$\begin{aligned} C_i &= \text{channel capacity of the } i\text{th channel} \\ d_i(C_i) &= \text{cost to supply } C_i \text{ units of capacity} \\ &\quad \text{to the } i\text{th channel} \\ D &= \text{total dollars available for design.} \end{aligned} \qquad (3)$$

This problem yielded a number of subcases, which could be solved exactly. Without going into the full set of cases, perhaps the most interesting was that of the pure channel capacity assignment which is the same problem as (3), but where we assume we are already given a routing procedure (i.e., a traffic flow assignment) and a topology. Then for the case of linear costs, namely, $d_i(C_i) = d_i C_i$, I was able to provide an exact solution for the optimal channel capacity assignment as follows:

$$C_i = \frac{\lambda_i}{\mu} + \left(\frac{D_e}{d_i}\right)\frac{\sqrt{\lambda_i d_i}}{\sum_{j=1}^{M}\sqrt{\lambda_j d_j}} \qquad i = 1, 2, \ldots, M \tag{4}$$

where

$$D_e = D - \sum_{i=1}^{M} \frac{\lambda_i d_i}{\mu}. \tag{5}$$

M is the number of channels in the network, and D is the total number of dollars available to provision these channels. I observed that this assignment allocated capacity such that each channel received at least λ_i/μ (which is the minimal required amount to keep the utilization from exceeding unity) and then allocated additional capacity to each node. Note that the cost incurred by assigning the minimum capacity to the ith channel is merely $\lambda_i d_i/\mu$ dollars, and if we sum over all channels we see that the total dollar allocation must exceed this sum if we are to achieve finite average delay in our network design. The difference between D, the total dollars available, and the minimum feasible allocation is exactly D_e, as given in Equation (5) above (we refer to this as the "excess dollars"). For stability, we require $D_e > 0$. From Equation (4), we see that these excess dollars are first normalized by the cost rate d_i and then distributed in proportion to the square root of the cost-weighted traffic $\lambda_i d_i$, over all channels; for this reason, I referred to this optimal capacity assignment as the "square root channel capacity assignment." If we substitute the expression for the optimum C_i back into our performance function as given in Equation (1) with the expression for T_i as given in Equation (2), we obtain the following result for the optimal (minimal) delay:

$$T = \frac{\bar{n}}{\mu D_e}\left[\sum_{i=1}^{M}\sqrt{\left(\frac{\lambda_i d_i}{\lambda}\right)}\right]^2 \tag{6}$$

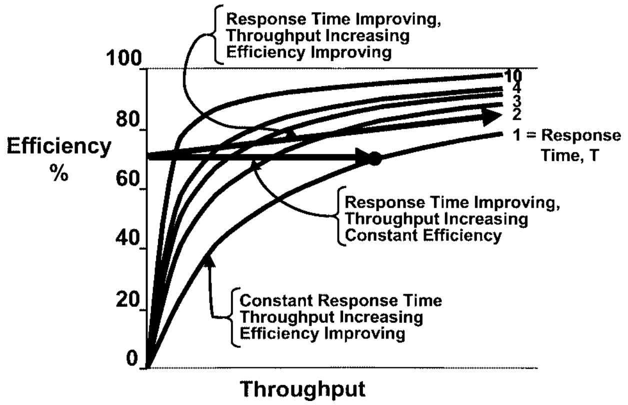
where $\lambda = \sum_{i=1}^{M} \lambda_i$ and $\bar{n} = \lambda/\gamma$ is the average path length traveled by messages. These equations represent the complete solution to the capacity assignment problem for the case of linear costs. From this solution, I was able to infer a number of properties of the optimal routing procedure and the optimal topology for data networks.

9. SOME PRINCIPLES

a. The First Resource Sharing Principle: The Smoothing Effect of a Large Population

Basically, this is the Law of Large Numbers. In terms of data networks, it can be articulated roughly as stating that (under a variety of conditions), whereas each individual traffic stream in a data network may behave in an unpredictable fashion, the merged behavior of a large population of traffic streams behaves in a predictable fashion. This predictable fashion presents a total traffic demand to the network, which is the sum of the *average* demands of each stream. Basically, this is the "smoothing" effect of a large population.

Figure 1. Key tradeoff: Response time, throughput, efficiency.

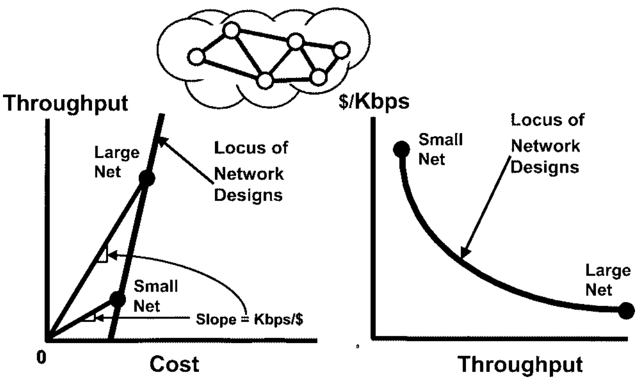


b. The Second Resource Sharing Principle: The Economy of Scale

I was able to show the following general result that I could apply to the design of data networks. Specifically, it said that if you scale up throughput and capacity by some factor F, while holding the average packet size constant, then you will reduce the average response time for that system by the same factor. Alternatively, if you scale capacity more slowly than throughput while holding the average response time constant, then the channel efficiency, i.e., the channel utilization factor, ρ, will increase (and can approach 100%). This alternative form of the principle appears to violate the basics of queueing theory, but I was able to show that it is correct and obviously has significant implications for network design. Figure 1 illustrates this principle.

One consequence of these principles became quite apparent once these results were applied to network designs. In particular, it was found that large networks displayed a true economy of scale in dollars as well as performance. The tradeoff between throughput and cost for a number of network designs is shown in Figure 2. We see the improvement in the cost per unit of throughput as the network size increases. It is immediately apparent that large networks have a distinct advantage, and this advantage derives from the two principles stated above.

Figure 2. Economy of scale in networks.



In my mind, from an analytical and algorithmic viewpoint, there are three basic components that made the Internet networking technology so powerful:

1. The key concept of demand access, i.e., dynamic resource sharing. Packet switching is one example.
2. The key concept of large shared systems. High speed channels is one example.
3. The key concept of distributed control. Distributed routing algorithms is one example.

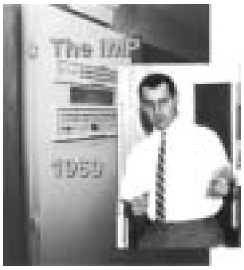
10. NO ONE CARED

Unfortunately, the commercial world was not ready for data networks, and my work lay dormant for most of the 1960s as I continued to publish my results on networking technology while at UCLA where I had joined the faculty in 1963. In the mid-1960s, the Advanced Research Projects Agency (ARPA)—which was created in 1958 as the United States' response to the Soviet Union's 1957 launch of Sputnik—became interested in networks. ARPA had been supporting a number of computer scientists around the country, and as new researchers were brought in they naturally asked ARPA to provide a computer on which they could do their research and moreover asked that their computers contain all the hardware and software capabilities of all the other supported computers. Rather than duplicating all these capabilities, ARPA reasoned that this community of scientists would be able to share these specialized and expensive computing resources if the computers were connected together by means of a data network. In 1966 ARPA enlisted the services of my former office mate at MIT, Lawrence G. Roberts (1974), to lead the effort to develop, manage, fund, and deploy this data network. It was largely through Larry's leadership and vision that this network came about. Because of my expertise in data networking, Larry called me to Washington to play a key role in preparing a functional specification for this network, which was to be called the ARPANET (1970)—a government-supported data network that would use the technology I had elucidated in my research, which by then had come to be known as "packet switching."

The specification for the ARPANET was prepared in 1968, and in January 1969 a Cambridge-based computer company, Bolt, Beranek and Newman (BBN), won the contract to design, implement, and deploy the ARPANET. It was their job to take the specification and develop a computer that could act as the switching node for the packet-switched ARPANET. BBN had selected a Honeywell minicomputer as the base on which they would build the switch.

Because of my role in establishing data networking technology over the preceding decade, ARPA decided that UCLA, under my leadership, would become the first node to join the ARPANET. This meant that the first switch, known as an Interface Message Processor (IMP), would arrive on the Labor Day weekend, 1969, and the UCLA team of 40 people that I organized would have to provide the ability to connect the first (host) computer to the IMP. This was a challenging task because no such connection had ever been attempted. (This minicomputer had just been released in 1968, and Honeywell displayed it at the 1968 Fall Joint Computer Conference, where I saw the machine suspended by its mounting hooks; while the IMP was running, there was this brute whacking it with a sledge hammer just to show it was robust. I suspect that this particular machine is the one delivered by BBN to UCLA.) As it turns out, BBN was running two weeks late (much to my delight, because my team and I badly needed the extra development time); BBN, however, shipped the IMP by airplane instead of by truck, and it arrived on time. Aware of the pending arrival date, we worked around the clock to meet the schedule. (See Figure 3 for a photo of the IMP and the author in 1969.)

Figure 3. The first IMP and the author—1969.



On September 2, 1969, the Tuesday after Labor Day, the circus began—everyone who had any imaginable excuse to be there, was there. My team and I were there; BBN was there; Honeywell was there; Scientific Data Systems was there (the UCLA host machine was an SDS machine); AT&T long lines was there (we were attaching to their network); GTE was there (they were the local telephone company); Larry and his folks from ARPA were there; the UCLA Computer Science Department administration was there; the UCLA campus administration was there; plus an army of Computer Science graduate students was there. Expectations and anxieties were high because everyone was concerned that their piece might fail. Fortunately, the teams had done their jobs well, and bits began moving between the UCLA computer and the IMP that same day. By the next day we had messages moving between the machines. Thus was born the ARPANET and the community, which has now become the Internet!

A month later, the second node was added (at Stanford Research Institute), and the first Host-to-Host message ever to be sent on the Internet was launched from UCLA. This occurred on October 29, 1969, when one of my programmers and I proceeded to "Login" to the SRI Host from the UCLA Host. The procedure was to type in "Log" and the system at SRI was set up to be clever enough to fill out the rest of the command, namely to add "in" thus creating the word "Login." The programmers at both ends each had a telephone headset so they could communicate by voice as the message was transmitted. At the UCLA end, we typed in the "L" and asked SRI if they received it; "got the L" came the voice reply. We typed in the "o," asked if they got it, and received "got the o." UCLA then typed in the "g" and the darned system crashed! This was quite a beginning. On the second attempt, it worked fine! So, the first message on the Internet was a "crash," but more accurately was the prescient word "Lo."

Little did we realize what we had created. Indeed, most of the ARPA-supported researchers were opposed to joining the network for fear that it would enable outsiders to load down their "private" computers. We had to convince them that joining would be a win-win situation for all concerned, and we managed to get reluctant agreement in the community. By December 1969, four sites were connected (UCLA, Stanford Research Institute, UC Santa Barbara, and the University of Utah), and UCLA was already conducting a series of extensive tests to debug the network. Indeed, UCLA served for many years as the ARPANET Network Measurement Center. (In one interesting experiment in the early 1970s, UCLA managed to control a geosynchronous satellite hovering over the Atlantic Ocean by sending messages through the ARPANET from California to an East Coast satellite dish.)

As head of the Center, my mission was to stress the network to its limits and, if possible, expose its faults by "crashing" the net; in those early days, we could bring the net down at will, each time identifying and repairing a serious network fault. Some of the faults we uncovered were given descriptive names like Christmas Lockup and Piggyback Lockup (Kleinrock 1976b). By mid-1970, 10 nodes were connected, spanning the USA. BBN designed the IMP to accommodate no more than 64 computers and only 1 network. Today, the Internet has millions of computers and networks! In 1972, electronic mail (e-mail) was an ad-hoc add-on to the network, and it immediately began to dominate network traffic; indeed, the network was already demonstrating its most attractive characteristic, namely, its ability to promote "people-to-people" interaction. The ARPANET began to be known as the Internet in the 1980s and was discovered by the commercial world in the late 1980s; today, the majority of the traffic on the Internet is from the commercial and consumer sectors, whereas it had earlier been dominated by the scientific research community. Indeed, few of us in those early days predicted how enormously successful data networking would become.

11. MY EARLY VISION

I am often asked if I realized back then how the network would evolve. The answer is both "yes" and "no." The "yes" part is more than a recollection, for indeed, I am quoted in a UCLA Press Release (Tugend 1969) that came out on July 3, 1969, a full two months before the Internet came to life. A copy of the press release is shown in Figure 4.

In that press release, I describe what the network would look like and what would be a typical application, and I am quoted in the final paragraph as saying, "As of now, computer networks are still in their infancy, but as they grow up and become more sophisticated, we will probably see the spread of 'computer utilities,' which, like present electric and telephone utilities, will service individual homes and offices across the country." In other words, I had the following vision of what the Internet would become. It would be:

a. ubiquitous
b. always accessible
c. always on
d. anyone could connect any device from any location
e. invisible

The "no" part of my answer is that I had no idea that my 94-year-old mother would be on the Internet today! That is, I did not foresee the pervasive impact of the Internet on so many people and on so many aspects of society and humanity. The first time I sensed this was when I saw e-mail sweep through the network in 1972 and then realized that the network was not about computers talking to each other, but rather it was about communities of people interacting.

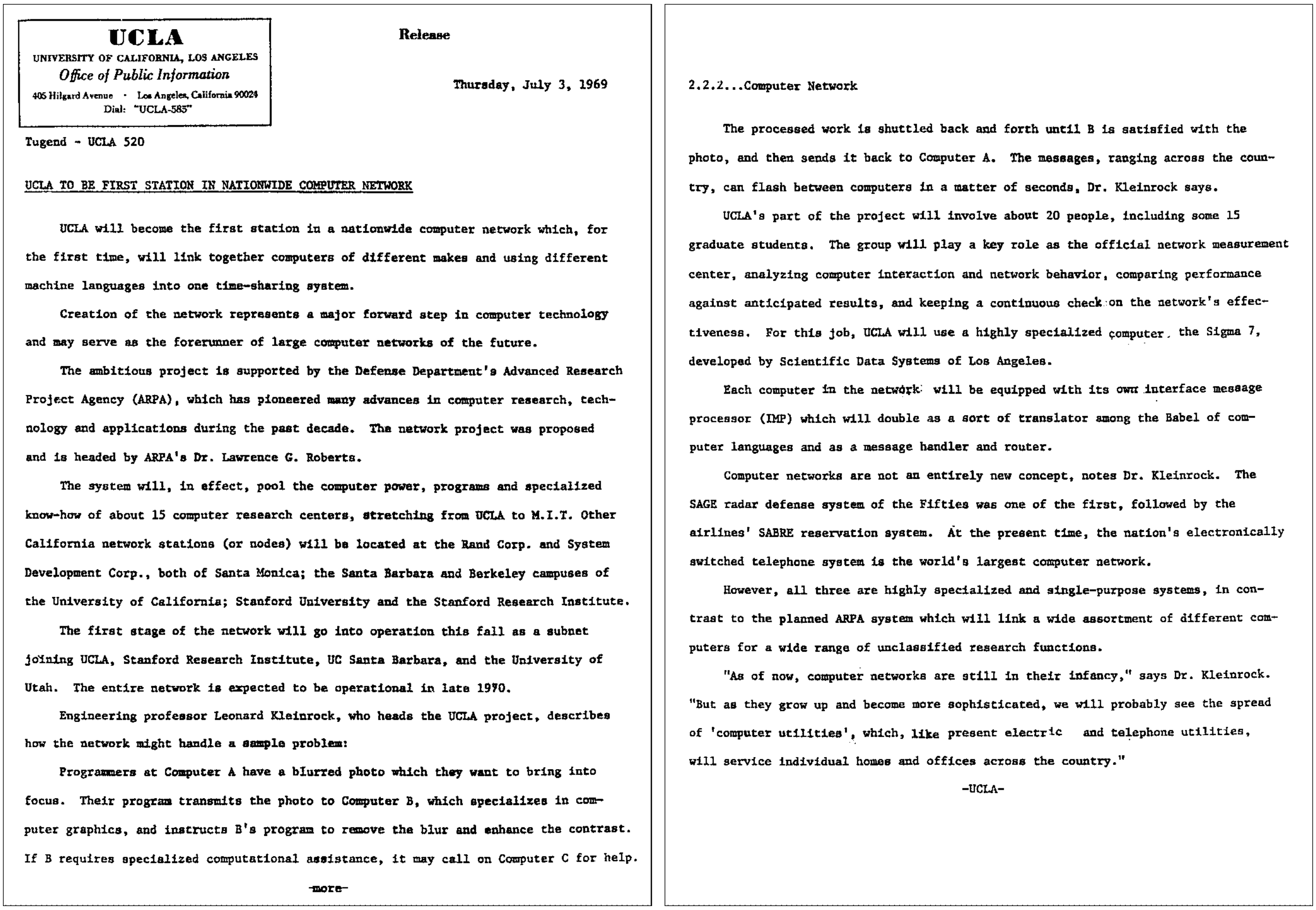
12. NOMADIC COMPUTING

The vision I articulated back in 1969 has not been fully realized. The Internet almost got it right, but not quite. The problem arises from the fact that in the early days of networking, it was assumed that a user, his IP address, his device and his location, were all intimately linked together. This was the mentality that gave rise to the highly successful and important set of protocols now referred to as TCP/IP. With that thinking, the first three elements of my vision were realized (ubiquity, always accessible, and always on). However the ability to move from one location to another with a computing device that was configured to operate in the original location was not, and is not still, easily accommodated at a new location where it is considered an "alien." The ability to appear at any location with any device and gain transparent access to Internet services is what has come to be known as "nomadic computing," and this is an active area of research, development, and product deployment today (Kleinrock 2000). In addition, the "invisibility" part of my vision implied that access would be as invisible as is electricity. (It's there, we don't have to think about it, and it serves us with a very simple and familiar interface.) No one today considers the computing and networking environment invisible; booting up Windows, configuring network parameters, establishing network connections, having personalized ubiquitous services delivered to individuals in a familiar and user-friendly fashion, etc., have a long way to go before they disappear sufficiently so as to be transparent and "invisible."

13. EPILOGUE

The potential impact of the ubiquitous information infrastructure of the Internet is unbounded. The nature of the services and styles it can produce is limited only by the imagination of its practitioners. I continue to be fascinated by the possibilities it offers and the fact that it permits the creativity of hundreds of millions of users worldwide to

Figure 4. The 1969 press release announcing the birth of the Internet.



UCLA
UNIVERSITY OF CALIFORNIA, LOS ANGELES
Office of Public Information
405 Hilgard Avenue · Los Angeles, California 90024
Dial: "UCLA-585"

Release

Thursday, July 3, 1969

Tugend - UCLA 520

UCLA TO BE FIRST STATION IN NATIONWIDE COMPUTER NETWORK

UCLA will become the first station in a nationwide computer network which, for the first time, will link together computers of different makes and using different machine languages into one time-sharing system.

Creation of the network represents a major forward step in computer technology and may serve as the forerunner of large computer networks of the future.

The ambitious project is supported by the Defense Department's Advanced Research Project Agency (ARPA), which has pioneered many advances in computer research, technology and applications during the past decade. The network project was proposed and is headed by ARPA's Dr. Lawrence G. Roberts.

The system will, in effect, pool the computer power, programs and specialized know-how of about 15 computer research centers, stretching from UCLA to M.I.T. Other California network stations (or nodes) will be located at the Rand Corp. and System Development Corp., both of Santa Monica; the Santa Barbara and Berkeley campuses of the University of California; Stanford University and the Stanford Research Institute.

The first stage of the network will go into operation this fall as a subnet joining UCLA, Stanford Research Institute, UC Santa Barbara, and the University of Utah. The entire network is expected to be operational in late 1970.

Engineering professor Leonard Kleinrock, who heads the UCLA project, describes how the network might handle a sample problem:

Programmers at Computer A have a blurred photo which they want to bring into focus. Their program transmits the photo to Computer B, which specializes in computer graphics, and instructs B's program to remove the blur and enhance the contrast. If B requires specialized computational assistance, it may call on Computer C for help.

-more-

2.2.2...Computer Network

The processed work is shuttled back and forth until B is satisfied with the photo, and then sends it back to Computer A. The messages, ranging across the country, can flash between computers in a matter of seconds, Dr. Kleinrock says.

UCLA's part of the project will involve about 20 people, including some 15 graduate students. The group will play a key role as the official network measurement center, analyzing computer interaction and network behavior, comparing performance against anticipated results, and keeping a continuous check on the network's effectiveness. For this job, UCLA will use a highly specialized computer, the Sigma 7, developed by Scientific Data Systems of Los Angeles.

Each computer in the network will be equipped with its own interface message processor (IMP) which will double as a sort of translator among the Babel of computer languages and as a message handler and router.

Computer networks are not an entirely new concept, notes Dr. Kleinrock. The SAGE radar defense system of the Fifties was one of the first, followed by the airlines' SABRE reservation system. At the present time, the nation's electronically switched telephone system is the world's largest computer network.

However, all three are highly specialized and single-purpose systems, in contrast to the planned ARPA system which will link a wide assortment of different computers for a wide range of unclassified research functions.

"As of now, computer networks are still in their infancy," says Dr. Kleinrock. "But as they grow up and become more sophisticated, we will probably see the spread of 'computer utilities', which, like present electric and telephone utilities, will service individual homes and offices across the country."

-UCLA-

contribute to its exponential growth and power. From my view, it has been a fascinating journey from a comic book to cyberspace.

REFERENCES

ARPANET. 1970. Set of five papers on the ARPANET. 1970. *Proc. Spring Joint Comput. Conf.* Atlantic City, NJ.

Boxma, O. 1979a. On a tandem queueing model with identical service times at both counters I. *Adv. Appl. Probab.* **11** 616–643.

——. 1979b. On a tandem queueing model with identical service times at both counters II. *Adv. Appl. Probab.* **11** 644–659.

Brockmeyer, E., H. L. Halstrom, A. Jensen. 1948. *The Life and Works of A. K. Erlang.* Danish Academy of Technical Science, 2.

Davies, D. 1973. *Communication Networks for Computers.* John Wiley & Sons, New York.

Hunt, G. C. 1957. Sequential arrays of waiting lines. *Oper. Res.* **4** 674–683.

Jackson, J. R. 1957. Networks of waiting lines. *Oper. Res.* **5** 518–521.

Kleinrock, L. 1961a. Information flow in large communication nets. Ph.D. thesis proposal, Massachusetts Institute of Technology, Cambridge, MA.

——. 1961b. Information flow in large communication nets. RLE Quart. Progress Report, Massachusetts Institute of Technology, Cambridge, MA.

——. 1962a. Information flow in large communication nets. RLE Quart. Progress Report, Massachusetts Institute of Technology, Cambridge, MA.

——. 1962b. Message delay in communication nets with storage. Ph.D. dissertation, Massachusetts Institute of Technology, Cambridge, MA.

——. 1964. *Communication Nets: Stochastic Message Flow and Delay.* McGraw-Hill, New York (reprinted by Dover Publications, 1972).

——. 1975 *Queueing Systems, Volume I: Theory.* Wiley Interscience, New York.

——. 1976a. *Queueing Systems, Volume II: Computer Applications.* Wiley Interscience, New York.

——. 1976b. ARPANET lessons. *Conference Record, Internat. Conf. Comm.* Philadelphia, PA, Jun 20-1–20-6.

——. 2000. Nomadic computing and smart spaces. *IEEE Internet Comput.* **4**(1) 52–53.

Morse, P. M. 1958. *Queues, Inventories and Maintenance.* John Wiley & Sons, New York.

Roberts, L. G. 1974. Data by the packet. *IEEE Spectrum* **11**(2) 46–51.

——. (1999). The first theory of packet networks 1959–1964. ⟨http://www.ziplink.net/~lroberts/SIGCOMM99_files/v3_document.html⟩.

Syski, R. 1960. *Introduction to Congestion in Telephone Systems.* Oliver & Boyd, Ltd. Edinburgh and London, U.K.

Tugend, T. 1969. UCLA to be first node in nationwide computer network: UCLA Office of Public Information Press Release, July 3. ⟨http://www.lk.cs.ucla.edu/LK/Bib/REPORT/press.htm⟩.