Oral History of Leonard Kleinrock

Interviewed by: Marc Weber

Edited by: Dag Spicer

Recorded: September 28, 2011
Los Angeles, California

CHM Reference number: X6295.2012

Weber: And I'm here today on September 23rd with Leonard Kleinrock, a major pioneer of networking. And thank you for doing this.

Kleinrock: My pleasure, my pleasure.

Weber: So really, just want to start with your background, childhood, maybe a little bit about your family background.

Kleinrock: Well, my parents were Polish immigrants. My father came here after the First World War, became a grocer. We lived in Manhattan. I was born in Manhattan. I was actually born in Harlem in a hospital that no longer exists, June 13, 1934. And I had a terrific childhood. We were pretty poor, but the environment in Manhattan was just so rich and so wonderful that it provided all the stimulation, entertainment, challenge that one could hope for. And I went to a really great elementary school, PS-173. Had some really strong classmates, and that helped challenge my learning and educational experience.

Toward the end of elementary school, I joined the Boy Scouts. And I was challenged by my Scout Master, after I rose to certain levels of the rankings, to become the first Eagle Scout in that troupe. And he felt I could do it, and no one else in the troupe had become an Eagle Scout, so I decided to take on the challenge. And a couple of years later, I was an Eagle Scout. And it seemed to me to be a challenge that was *just* almost within reach, and it required a lot of effort. Had to go to Scout camp a couple of times. Bird Study was one of the merit badges, which in New York City is a problem. As a kid I just thought there were only three birds in New York, a sparrow, the pigeon and the seagull. Turns out there's a huge number that migrate through Central Park. I learned that later. So I had to go to Scout Camp to see the 40 birds, and whatever, and I became an Eagle Scout.

Weber: The camp was out of the City, right?

Kleinrock: It was out of the City, absolutely. It was a tough camp. You know, they don't pamper you in the Scout Camp, let me tell you. But the reason I'm mentioning that is because it was my first success was clearly-- you can mark it as a success-- it was a challenge, I accepted it. It gave me recognition. But mostly, I recognized for myself that I was able to achieve *just* beyond my grasp if I set my mind to it. And that stuck with me my whole life. The fact that I could achieve success if I just set out to do it, and if I have a challenge that was worthwhile in my mind. From there, I went on to Bronx [High School of] Science.

It was a terrific education, but even before elementary school, as I was entering elementary school, I was reading in a comic book-- I loved comic books-- and in the centerfold of this Superman comic book was a description of how to build a crystal radio. I had no idea what a crystal radio was. What fascinated me was, you could make it out of parts you could find around your home. So I needed something called a "crystal." But you could make that out of your father's used razor blade and a piece of pencil lead. That was easy to find. And I needed some wire. Well, I found wire in the neighborhood on the streets. You needed an empty toilet paper roll. That was easy. And then I needed an earphone. An earphone was a sophisticated thing, but I knew where I could get one. In the candy store downstairs in the street, I could steal it from the public telephone. And I did. I stole a damned earphone from the public telephone. And now I needed one more part. It was called a variable capacitor. And I knew I couldn't find that anywhere in the neighborhood, but I knew where I could get one, and it was down in the radio row, Canal Street, in Lower Manhattan.

So my mother took me down in the subway, walked into the first electronics store, banged my fist on the table and I said, "I need a variable capacitor." And the clerk behind the counter said, "What size?" <laughs> I had no idea. It blew my cover and my bravado. And I explained to him why I wanted it. He says, "I know just what you need." Went in the back, got this variable capacitor. Sold it to us for a nickel. That was the only expense I had. Took it home, wired it up, and I could hear music out of that earphone! No battery, no power. It was magic! And the magic attracted me. I wanted to understand how this works. I'm still trying to understand how it works. It is actually a mystery, the idea of electromagnetism and suchlike. But that got me started. Little did I know an engineer was born that day! And I so I began to tinker around with radios.

Weber: And how old were you then?

Kleinrock: I was just over six when this happened.

Weber: Wow.

Kleinrock: So I began to collect old radios, broken down radios. I'd cannibalize them. Try to put them together and make a new radio. I began to read about radio design, and I just had a wonderful time. When I got to Bronx Science in high school, I took as one of my electives, a radio electronics course, and that was amazing! It opened my to wonderful new technical studies of this thing. Meanwhile, I had gotten a book out of the library. I remember the book exactly. It was by Marcus, Marcus and Horton, called "Elements of Radio." And not only did it have a bit of mathematics, which I couldn't understand when I was a kid. But it described what was going on. You heat up a filament, the electrons boil off, they're in a vacuum, they get collected. And it was so intuitively descriptive, just caught my attention. And from there, I learned how to do some electronics. So in high school, I studied more electronics, as well as the regular curriculum. And I decided I'm going to get an electrical engineering degree.

Weber: And your parents were encouraging?

Kleinrock: My parents had no science background at all. Some years later when the *Miami Herald* interviewed my parents, my mother said, "It was all Greek to me." She had no idea what I was doing, and understandably. But they, of course, were very much encouraging for education and achievement. We were quite poor, and I always felt like one of the poorer kids in the neighborhood. I lived on the wrong side of Broadway. I lived in a very tough neighborhood for a while, and was the only Jewish kid in the neighborhood, and so that made it especially difficult.

Weber: Do you remember the address you grew up at?

Kleinrock: Well, the place where I went to elementary school was basically 601 West Broadway. And the place where I went to college in, and later high school, was 1253 Saint Nicholas Avenue, between 172nd and 173rd in Manhattan. So I decided I wanted to go to college, of course. I applied to every Chamber of Commerce in the United States to get a scholarship so I could go to an out-of-town college. And I got lots of scholarships, but they weren't enough. Because some years prior to this, my father had gotten very ill. He'd become a grocer. He had to sell his grocery store. And he couldn't work anymore. So I had to bring money into the house. It wasn't enough that I didn't cost anything.

So I was ready to go to City College in New York, CCNY in September of 1951, when I realized I couldn’t afford to do that. I needed to bring money in. So I got a job in Downtown Manhattan in an electronics firm. And I went to night school. And this was actually a rather interesting, in many ways a fortunate, turn of events. Because I was working in exactly the field I wanted to, while learning about it in college. And the evening session instructors were folks who were in the daytime working in the field. So they brought not only the theoretical understanding, but the practical knowledge. I'll never forget, toward the end of my undergraduate education, transistors were just starting to come into vogue. And our professor came in and he held up a transistor, and he said, "See this transistor? It's a better thermometer than it is an amplifier." Because it was so heat sensitive. And you had to take basically measures to counteract the effect of the heat. Vary amplification with temperature change. You never would have learned that from a daytime professor! So it was a rich-- so the guys and gals in evening session was another collection of people. You can imagine. A lot of misfits, a lot of poor folks struggling, a lot of crazies, gangsters. But that enrich-- and by the way, this was shortly after World War II. I entered college in 1951, a few years after World War II. Lot of GI Bill. These guys came back with *life* experience, as did the fellahs I was working with in the firm down in lower Manhattan. So it was a really enriching experience. It wasn't kind of a straight arrow, narrow world that so many of my other classmates lived in in daytime.

Weber: People of all ages, right?

Kleinrock: All ages. I mean, and you just worked really, really hard. I did most of my studying on the subway, commuting down to work and back to school. I would take an 8-1/2 x 11 piece of paper, fold it in half, fold it in half again, get these narrow columns, put the critical facts on the paper, and study like this on the subway, because you didn't have enough room to do that standing up. It was a good experience.

Weber: Now, did you have brothers and sisters?

Kleinrock: Yes, I have one sister. She's two-and-a-half years older than I am. She is not in the sciences. She, in fact, is a psychotherapist. We fought like hell. When I was young, I hit her over the head with a hammer. We had fights. So we didn't talk very much. <laughs> But I really had a great experience as a kid growing up. I built the radios I was telling you about. Model airplanes. And I was totally engaged in experimenting with projects and things and building and taking things apart and putting them together. Because I really wanted to understand the way things work. But the interesting this is there was no external scientific heritage. Except now that I look back at my father's family and his relatives. There are doctors, there are engineers, they were scientists. Men of great accomplishment. So there was a kind of basis there, but no direct experience.

Weber: And did you have friends that shared your interest in these sorts of things?

Kleinrock: The answer is "no." In Bronx Science, yes, some-- there were other folks in my radio or electronics class. But they didn't live near me. I had to commute to get to Bronx Science. In the neighborhood I never found folks that were participating. The kids from school were interested in softball, which I was also, but they spent their time studying the batting records of the local hardball teams, and the professionals. That never interested me. And so I'd play sports, which I loved. But I didn't want to spend my extra time there. And after all, I had to get to Hebrew school at 4:00 every afternoon until I got Bar Mitzvah-ed. Had to play the violin an hour a day, which I hated. You know, all the extra shackles, it did limit my social experience. I had some friends, but they were not engaged in the activities in which I was really very much interested in.

Weber: So you felt that was very much a life-- sort of an inner life apart.

Kleinrock: It was. It was. And it was a life in which I had enormous gratification. In the Boy Scouts, with the Eagle Scout, and even before, I rose through the ranks of leadership. You know, Boy Scouts taught me an enormous amount about dealing with people, leading them. I became a Senior Patrol Leader, a Junior Assistant Scout Master. And that was, again, another dimension of growing up, which was much wider than just a pure narrow educational tract. But I got to tell you, those guys down in the company I worked-- it was called Photobell Company, Incorporated, it was an industrial electronics firm-- we built photoelectric devices that would detect things. Cereal boxes moving on a conveyor belt, rocks piling up in a rock pile. And the thing about that particular set of applications is either there's something there, or there isn't. And the electronics of the time was using analog circuitry, and when it crossed a threshold, you'd say, "Ah, I've detected something." So I'd just learned about digital electronics in college around this time, and I said, "Let's make a binary system here. A multivibrator, a flip-flop." And that was perfect. Either it's there or it's not. And it's a very sharp and detectable threshold. So it was nice to be able to introduce that kind of experience from school into my work experience and vice-versa.

Weber: Because anything that's not a clear signal is just a noise.

Kleinrock: It's noise-- in the case of the binary circuit. In case of the continuous analog, you're not quite sure where to set the threshold, and it could bounce back and forth across that threshold, and cause all kinds of disturbance. This was a rock-solid decision. It was very, basically, gratifying to be able to introduce that into the firm's product line.

Weber: So you were the only person who was probably keeping up with that? You were getting it in school.

Kleinrock: Well, it was brand new stuff, and so I could introduce this new technology, digital electronics, when all the other really great analog electronic designers, who taught me a great deal while I was going to school, were not yet caught up with the digital world.

Weber: So how did that change your status in the firm?

Kleinrock: Well, I rose through the firm. I started as a technician. In the middle of my undergraduate work, I became an engineer, an assistant engineer. And when I graduated, I went to graduate school. So I took on various roles, but there was no step change there. I also got married while I was in undergraduate school. I was just under 20. My wife was over 19, particular over 18. So she could get married without her parents' permission, and I needed my parents-- men under the age of 21, needed their permission.

Weber: But your parents had no problem?

Kleinrock: Oh, no. Well, yes, of course, they did. They didn't want me to get married at that age. But what is a 20-year-old? You can't convince them.

Weber: How did you meet her?

Kleinrock: I met her at a dance, related to a college event.

Weber: Which college did you? City College?

Kleinrock: CCNY in Upper Manhattan.

Weber: So she was going to the night school as well?

Kleinrock: No, no, no. She was in high school when I met her. And then she went to Jackson College of Tufts University for her first semester. But we got so involved, she came back and then went on to get a degree. And so I was going at night, sending her through college, feeding my parents money, and finishing my school at night.

Weber: And so that took longer than four years, obviously, going to night school.

Kleinrock: Yeah, I went at basically 3/4 rate. And every summer, and it took five-and-a-half years, because Double E, Electrical Engineering, was more than a four-year curriculum in those days. It was usually four-and-a-half or more. So I went to the maximum pace I could given the hours I could devote to attending class. It was a struggle. It was, you know, it was-- here I was ready to go to CCNY, just enjoying all my training in Bronx Science, and then I'm put into this black tunnel called "evening session," from which the dropout rate is huge! And the duration could be huge as well. So I always had the, you know, the question, "I got to get through this thing!" And it took a lot of perseverance.

Weber: And you were living with your parents?

Kleinrock: With my parents until I got married. And in fact, I think now-- I realize now that's part of the motivation that I got married so early. It was a confining environment, it was struggling, and I needed to spread my wings and get some freedom and some independence.

Weber: So then you moved in with your wife.

Kleinrock: Yeah, we took an apartment in West Bronx. Stayed there. And then toward the end of my undergraduate curriculum and degree, it was announced that someone was coming from MIT Lincoln Laboratory to discuss this scholarship program that MIT was making available. And was going to speak at 4:00 in the afternoon, so I asked my boss if I could take off early that day, and made it up another day. And I went to hear this presentation. And this guy described something from MIT Lincoln Laboratory that was amazing. It was called the Staff Associate Program. They would pay you as a research assistant while you were there during the semester for two semesters. You'd work at Lincoln Laboratory full-time in the summers. Third semester, you'd be a full-time student, full pay, even without being a research assistant. In the fourth semester, you should have finished your master's thesis by then, and have one course left, and commute back and forth to Lincoln Lab and MIT for the one course. It was a *terrific* program! And so I really got excited by this. And the man speaking there at City College said, "If you want an application, see the professor in the back of the room." So after the lecture, I went back there, and I said, "I'd like to have an application." And the professor said, "Well, who are you? I don't recognize you." I gave him my name, and I said, "I go to evening session." He says, "Evening session! Get out of here!"

And I won't tell you what his name is for the record, I won't, but his name fits his personality. It's apt so often-- it's a nasty name and it worked. So what I did was I wrote away to Lincoln Lab, got the application, filled it out and was accepted in this program. No one else from City College was accepted, by the way.

Weber: Including all the others from the day school.

Kleinrock: Right. By the way, I was top student in day and evening. And I was a student body person for evening session. So the credentials were there. And MIT had the habit of taking sort of the best student from many colleges around the country, which made an interesting mix. And so off I went to get my master’s degree. Only the master’s degree to MIT.

Weber: And your wife was in college then?

Kleinrock: She finished at the same time. ’Cause I took the five-and-a-half. She was a year-and-a-half younger than I am. She finished in four years. And we went up there. She got a job as a medical technician in the Harvard Medical School.

Weber: What did she study?

Kleinrock: She studied physiology. And wasn't too long after that that we had our first child. In fact, we planned it. I told you the schedule. After the third semester, you work at Lincoln Lab in the summer, and then you commute. And you're getting a full-time salary when you commute. Okay, we're going to have the first child right there in that summer. And sure enough our first child, Marty got born in August of 1958. I started this program in '57. Interesting, when I got-- in January '57. What's interesting, by the way, I accelerated in high school so that I would end up a June graduate from high school to fit the schedule of universities and colleges. But it took me five-and-a-half years to get through City College, so I was now out of phase again, so I ended in January.

But my first boss at Lincoln Lab, the first summer, was a gentleman named Ken Olsen. I'm sure you recognize he's the gentleman who started the Digital Equipment Corporation. Well, 1957 was when this was. The Summer of '57. That's the summer he formed DEC, and he asked me to join him, and I said, "No, I can't. I'm on this great program. I have to get my master's degree." And I never regretted that decision, of course. But he was a terrific guy to work for. You'd build something for him, and he'd look at the performance and say, "Try a little more. Make it a little better." He'd never [be] harsh, direct. He just would coax you to improve it. So I built for him something called a variable pulse delay amplifier. You'd dial in the time 100 milliseconds, click it, 100 milliseconds later, a pulse comes out. That was one of the first things DEC marketed. They didn't market computers in the beginning. They marketed modules from which people would build digital devices. And thereafter they begun to build their DEC machines and all the rest. The PDPs first. So anyway, that was a very interesting experience. And toward the end of my master's degree, I'd done a master's thesis on something called the Kerr-magneto optic effect. I'm sure you never heard of that.

Kleinrock: I mean, this is one of those long answers, huh? I didn't really go linearly through my childhood, 'cause I jumped around a bit. But does that work for you?

Weber: Yeah, that's fine. Unless there's something major that you skipped?

Kleinrock: Well, the fact that my father got ill, and I got in-- because that was really a very important point. I was 11 years old when that happened.

Weber: Sure, tell that.

Kleinrock: So I was in elementary school one day. My father was a successful grocer. He had his own store. And then he got seriously ill with asthma. Couldn't continue to work. Had to sell his store. And he didn't work for a number of years. And so you can imagine the family economics just crashed. My mother went to work, and I was working part-time all the time as a student from elementary school. I was an usher, I was a lifeguard, I was a clerk. And it was a big change in my life, because it was clear as I approached college that they couldn't afford to send me or to support me, I had to support them. And so I started contributing to the house at that point.

Weber: What did your mother do when she worked?

Kleinrock: Wonderful! Interesting story about my mother. When I was a little kid, she had been a typist, a secretary. And she went to vocational school. Instead of the academic track she took the vocational track in high school in New York City. And she was a terrific typist and secretary. Of course, when she had the children she stopped working. When my dad got ill, she took in typing to generate a little money at first. And I remember she would type envelope address-- addresses on envelopes. Thousand envelopes, five dollars.

But I watched this woman type, and it was fascinating. I mean, she was really good! So I taught myself to type. Thereafter, we needed more money, and so she went to work in Barton's Candy Stores. And she rose up to the level of manager very quickly, and she ran her stores. And any time I visited, you can imagine she would give wonderful samples of this delicious candy! The turtles, in particular, chocolate turtles. So I visited her when I could. And then after that, she took a job in a firm and became the person in charge of their business records. And she actually began to use some automatic data processing systems before she and my father went to live down in Florida.

Weber: Based on punched cards?

Kleinrock: No, it was some form of tape at the time. It was not a standing computer at all. It was a very specialized computer for some business processing. I think UMM was the firm that was involved with that. I never visited her there.

Weber: Did your father work again?

Kleinrock: Yes, my father, after many years lying in bed, taking all kinds of medications, he found an individual, who was a vegetarian, and through the power of this man's personality, brought my father out of it. And the vegetarian diet helped a lot. You know, cutting out milk products. Anything that caused congestion. And then he began to work as a clerk, which was real step down for him. 'Cause I saw him as owning his own store. And this guy was terrific. He would wait on three or four customers at the same time. He'd charm them, he'd engage them. He was zipping around. In the old days with the paper bags, he'd put the groceries in. You'd write the prices on the paper bag, you'd add it up, and that's what you'd pay. He'd right it down and go zip! <cluck> Zip! <cluck> Bang. I mean, again, really impressive guy.

So he began to work as a clerk. And I thought that was a real step down for him. He never complained about it. He worked long hours. But he never was able to come back up from that crash, if you will. And then some years later, he was very, very anxious to move down to Florida. And so he and my mom went down there, and we got them a condo, and they lived there. They lived quite a while. My dad died at the age of 94. My mother died two months shy of 100. Two months shy of 100. And she was using the Internet. And that gave me great pleasure! While she was alive and even now that's she's passed away-- she passed away in 2007-- I make sure to make the point that everyone uses the Internet, even my 99-year-old mother used the Internet. And it's impressive. She was using, you know, simple forms, but it was very, very gratifying for me that she could do that.

So as I was finishing my master's degree, I did this work on the Kerr-magneto optic effect. This was an effect that if you shine polarized light onto a magnetic surface, it'll rotate the polarization according to which way the surface is magnetized. Now there's a similar effect called a Faraday effect, where if you shine polarized light through a magnetic field, it'll rotate the polarization. And that was perfect for reading out of a thin-film memory. Small films of magnetic material, a one or a zero, instead of sensing it the way they had to in the past, you could do it with shining a light. So I built an apparatus and studied the effect, enhanced it and analyzed it and actually developed [it] way before anybody had some optical logic. Because you go bounce-bounce-bounce to different-- you can institute a logical function. That was never built. The problem was the effect was pretty small. So you could read it, but it never became practicable.

Weber: Get lost in the noise.

Kleinrock: Yeah, it would get lost in the noise, and you needed a strong magnetic, a thin film was a little too-- a little magnetization. So my professor, when he knew I was finishing up, said, "You have to go on for PhD." I said, "No, I've organized this. I've got this staff associate program. My child was born in August. You know, I got this plan." He says, "You have to get a PhD." And he twisted my arm, and I spoke to my wife, and I thought a lot about it. I spoke to my colleagues, and I decided, "Okay, I'm going to do it."

Weber: But with the master's your plan was then to go on to what?

Kleinrock: With the master's I was simply going to become a researcher at Lincoln Laboratory. And by the way, the wonderful part about the Lincoln program, they didn't put any pressure on you. They didn't even expect you to work for them afterwards. And I'll tell you a little later on what happened some years later about the employment there. But I was just going to go there, master's degree and become an engineer. But the pressure was on, so I decided, "I'm going to do it." And this is an important part of the story, because it set my mind as to how I was going to approach this. I said, "If I'm going to spend another three or four years getting a PhD, I want to do something meaningful, not some narrow, small mathematics thing. I want to get something that's going to have impact on the world." And I also said, "I'm going to work for the absolute best professor at MIT that I know of." The best professor at the time, in my opinion, was Claude Shannon. Giant of a man intellectually. Brilliant. And he'd gotten there in 1958, I believe, and this was 1959 when I was considering going on for a PhD. So I called him up, and I said I'd like to work for him. Well, he didn't know me from Adam. He says, "Come, I'd like to meet you. Do you mind coming to my home this Saturday?" I said, "You bet!" <laughs>

So I told my friends at Lincoln Lab, "I'm going to be visiting Shannon this weekend." And during the week, Shannon called me up and said, "Look, I hope you don't mind. I have a house-guest. Jan Peerce, and he's staying with me. He'll be there when you come." I said, "That's fine." What am I? A little pipsqueak. I'm not going to tell him, "No." I was very happy to come. So I told my colleagues about this, and what I heard Shannon say, I heard him say, "Jan Peerce." So I told my colleagues, and they said, "Well, look, when you go there, ask him to sing the "Bluebird of Happiness."" He was the great opera singer, Jan Peerce. When I got there, I didn't find a rotund opera singer, I found a skinny, very unpleasant man, John Pierce. He was a great, brilliant guy from Bell Labs where Shannon had worked. And here I am, this was after my son was born. And I'm bubbling over with the feeling about this new child. "Oh!" Talking to Shannon and Pierce about this. And Pierce says, "Ach! Children! They're terrible! You'll give them everything. They'll disappoint you!" He was a real downer! And here I am bubbling over. This guy try to-- it didn't work, of course. I was enthusiastic all the way. I spoke to Shannon, and he agreed to take me on.

Weber: What was Shannon like?

Kleinrock: Oh, Shannon was a wonderful guy. Very approachable. Not a jolly guy. Intense, but open. And you know, no pretense or airs. He sat me on his porch, and he was sitting in a chair, and I was sitting here. And as I was discussing with him, I noticed his eyes going like this. Turns out behind me was a hammock. And I don't know who it was, but someone, they were swinging the hammock and the thick bar, wooden bar of the hammock was coming within inches of my head. Had I leaned back, I would have gotten a whack in the head! <laughs> So Shannon didn't tell me about it. He just, he was looking. He showed me his automatic lawn mower. He had installed a wire in his lawn, and this lawn mower would go around and automatically mowing. Wonderful guy. And I got to know him pretty well over the next few years.

Weber: He had just moved up...

Kleinrock: About a year before that. And he was, again, very focused, very direct. And what he wanted me to work on was a chess-playing program that he was developing. It was the first MIT chess program. And he was working with John McCarthy on it. Now we had some wonderful people at MIT at the time. So McCarthy and his student, Paul Abrahams, they wanted to take on the job, the programming job of generating legal moves. They were going to generate legal moves given the position of the board. Shannon, bless him, was interested in the strategy. How do we effect an appropriate strategy, given the position? And that was my part with Shannon. So Shannon handed me a book by a fellow named Fred Reinfeld, who was the great chess book writer. And this book was called, "1,001 Brilliant Chess Sacrifices and Moves." And each page had a situation, and it would say, "Black has a brilliant sequence. Find it." Next page, different situation. And of course, it was very hard! So Shannon said, "Go to the back of the book where the answers are. Look them up, and find out which is the most common first move of a brilliant sequence. That's something that's important for our program to know!" "What's the second most brilliant move? What's the most frequent second move?"

Weber: That quickly becomes...

Kleinrock: It does open up, but it limits the things you're going to do. You don't worry about position here and there. If there's a brilliant sequence, you have to go after it. So guess what is the most common first move of a brilliant sequence?

Weber: Moving a pawn forward.

Kleinrock: Well, that's a very weak move, you see?

Weber: <laughs>

Kleinrock: Check! Not the first move of the game, but in the middle of the game. We wanted to do the middle game. Openings are classic. Endings are sort of classic. The middle game is where the richness is. Well, "check" does this to the opponent, and now you can take the tree much deeper. Second most common move was a capture. A fork, a pin. Where you basically shackle the opponent. Anyway, so we did this, and we wrote this program, and it worked, and meanwhile, I realized I don't want to do that for my PhD dissertation. It's very interesting, and I enjoyed working with Shannon. But what was in vogue at the time was Shannon's great work on information theory and coding theory. And I felt that's probably where I would work. But I was taking courses at the time. I took some courses from Shannon as well. And I saw my classmates and the more senior students. And most of them were working for Shannon, or Shannon's disciples. And it was pretty clear to me at that time that most of the good work was already done by Shannon. The low-hanging fruit, the really impactful work was there in his 1948 classic paper. And these brilliant students were working on some interesting problems. They were all very tough problems. And they were-- in my mind-- of small consequence. And I said, "That's not what I signed up for."

Weber: And what courses was he teaching?

Kleinrock: He was teaching "Information Theory."

Weber: Okay.

Kleinrock: He was an interesting teacher, by the way. He developed the courses on his own. And he would use… he would use something called a "homely example" of coding and errors. I mean, he would talk about chopping logs in the north woods of America. And you got to ship them to the lumber mill. But the boxcars are too small to take the whole log. So you cut them up. So they can get there. But you want to put them back together-- and send them – this is imaginary -- to put them back together when you get there, what do you do? You have to label them. So the analogy, what if someone changed the labels a little? But that's noise. What do you do? So these homely examples were really very effective. He couldn't do arithmetic very well. If he had to multiply two numbers in an example, he would write them out and do long multiplication. There was nothing pretentious about this man. And you know he was a great juggler and unicycler at Bell Labs. And when he was lecturing, he would take the chalk and while he was answering the question or talking, he'd flip the chalk and catch it. You know, over here. You know, his juggling capabilities came out. It was a nervous twitch, but it was fascinating to watch.

Anyway, you'd walk into his office. This great mathematician, this great intuitive information theorist, what is he doing? He's got a Swiss army knife and a differential gear, and he's unscrewing the damn thing. He was very mechanically oriented. He built, I think, what was the first bus for consumer use, for personal use. He took a school bus, and converted into a thing that he could drive around in and sleep in. I mean, and that practical side of him was really the intuitive side, which matched the theoretical side, and when you could put those two together, the power of that combination is unbeatable. And very few people have that ability. Either they're pure theoreticians, or pure engineers, build, make things. But he could-- he understood the intuition of the mathematics. I remember one time-- he was on my committee, and I brought him a solution to a problem. And I brought it to him because the solution was extremely simple for a fairly complex problem. And I couldn't understand how could it be so simple. You know? It must be-- it's trying to tell me something. So I went to Shannon, in a minute, he explained exactly how the intuition predicted that answer. And now it all became clear. I mean, and he was always like that. Just able to extract out of a problem the essence. And that was an important lesson for me, because that's what I do as a researcher, and that's what I teach my students to do. And that's the big gripe I have about research today. And let me answer that right now. Much of the work you see presented at conferences in the performance evaluation area is the result of a simulation.

They give you a problem, a wireless network problem, a mission at work, a computer performance problem, and they'll simulate, and they'll show you the performance. Here's the way it behaves. And they'll say, "There's my answer." And my point is, "That's not the answer at all! That's a particular case, but if you see a curve going up, and flattening out, what does the slope of that line mean? Where does it come from? If it flattens out here, why there? What's the ascend static value, analytically, in terms of the parameters, not the particular numbers you put in? And if you don't understand that, you haven't learned anything." Well, Shannon was able to make that match all the time. And he taught me to seek that, and I became actually fairly good at it. Because I always asked myself, "What is this result telling me? What does it mean? How does it generalize?" Anyway, let's see, where was I?

Weber: But in a way he thought, obviously, this was not your personal experience, but he was somewhat of a misfit at Bell Labs, as well, right?

Kleinrock: A misfit-- everyone was a misfit there. All the greats were misfits. He had his idiosyncrasy of bicycling on the unicycle up and down the halls, and down the steps. But each of them had their quirks. I mean, look at the people that were there. I mean, and the work that came out of there. Richard Hamming was as quirky as anyone. And Steve Rice, and McDonald-- all those people. They were unique characters. And Bell Labs encouraged that kind of mind and intellect to come in and interact. That's gone now. That's one of the great losses of today.

Weber: When he came to MIT, he fit in well? He became part of that...

Kleinrock: Well, when he came to MIT, when I saw him, he was teaching. He took on very few students, very few. And, notice, he went into chess. He went into artificial intelligence. One lecture that he and... I guess it was Minsky gave to the graduate students… You know, one other thing about MIT, they'd be these colloquia, and these greats, the Shannons and the Minskys and the McCarthys and the Huffmans and they would get out there and talk to you, and they'd describe a project they wanted to get involved with. So I remember Shannon got up there along with Minsky. They wanted to build an artificial hand.

And, again, an example of Shannon's wonderful insight, Minsky was talking about the software and some of the pulleys, maybe. Shannon got up and he said, "Look at the way the arm moves. It moves like this. It doesn't go like this, as a robot would. You'd build a... You have to feed yourself. This is an evolutionary trait in mammals." I mean, that kind of observation was, you know, he just saw things the right way. So he branched out into much more than just information theory, and he preferred to supervise people not working in information theory. He had his disciples take on many of those people. But, for example, Ivan Sutherland was a student, Sketchpad. I mean, that wasn't information theory. It was graphical user interface. It was object-oriented programming and all the rest.

So you ask, how did Shannon fit in? He fit in okay. He was a good teacher, didn't teach too many courses. It was a privilege to work with him. Norbert Wiener was there at the same time. And those are the two sources that we now call 'communication theory.' You know, just to digress for a moment again, Shannon and Weiner both did some great work during World War II. They were both interested in combatting noise. And Shannon worked in cryptography and communications. Weiner was interested in automatic fire control, predicting where the object will be. And in both cases, you get a signal with noise in it, and Wiener was very good at eliminating noise, the Wiener-Lee filter and all the filtering and extrapolation and all the rest.

Shannon was interested in cryptography. But they both came to the same conclusion independently, and it's a very wonderful insight, again. At that time, it was clear that noise should properly be described statistically. You can't predict it, typically. They both recognized that the signal you were trying to send should also be described statistically, and that's surprising, but when you think about it, if there's no uncertainty in your mind as to what I'm going to send you as a signal, then why should I send it? If you know what it's going to be? So, it's a probabilistic event. Brilliant observation! Now, Shannon… well Weiner took that and said, "Okay, we're going to work and we're going to play around with the signal I receive and by very clever means extract the signal-plus-noise and pull the signal out of it.

Weber: What you just said sounds a bit like when he assigned you the chess.

Kleinrock: Well, in a way. There was a signal in the results.

Weber: But this sort of probabilistic, what's the most likely...

Kleinrock: Yeah, probability was the key to his thinking. Now, I'll describe that in a moment, but Shannon added one more piece. He said, "Not only is the thing I receive statistical and has to be worked on, but I can prepare the thing I send. I can doctor it up so it's best suitable for the noise it's likely to combat." That's called coding theory, redundancy. You add extra bits. They tell you something about what happened... So they went different ways, but what I'm saying is, Weiner was also a very strange lecturer at the time. He was really quite paranoid. But Shannon was a good lecturer, but he didn't stick to information theory, per se, and he didn't take on many students. But, again, the thing I observed is that many students working in information theory and coding theory were not doing the kinds of things I was interested in doing.

Now, at Lincoln Lab and at MIT, I was surrounded by computers and they couldn't communicate with each other and I knew that sooner or later, they're going to have to talk to each other. This would be a big thing. It's going to revolutionize where we're heading in this world of technology. And no one was working on that problem, and I had an approach to it, so this was perfect for me: low-hanging fruit, I think I see how to do it, it's a hard challenge, if I do it right, it will have meaning. So I set off to do it.

Weber: So people like Shannon and Weiner working on information, surrounded by computers, but they didn't explicitly set that as their goal?

Kleinrock: They never articulated that, no. But Shannon was on my committee, by the way. He was very happy to be on my committee, but he was not my supervisor. My supervisor was a fellow professor, Ed Arthurs, who was, again, a brilliant guy who had thousands of ideas, half of which were good and half of which were not good. The trick was to filter out which half. But he was interested in this problem because he had been working on a classified study called the Barnstable Study about a military-classified project which he couldn't tell me about, but the concept was communication of data, having computers talk to each other. So I recognized this was a problem to work on, I set out to do it, and I launched the research in that area.

Weber: And how did people react to that?

Kleinrock: Okay, so, people didn't recognize that this was an important problem. Again, Shannon thought it was good to work on. Arthurs felt it was important. And the third member of my committee I got out of the operations/research area at the Sloan School, a fellow named Herb Gallagher, who knew something about queueing theory, because I recognized the tool I needed to evaluate data networks has to talk about throughput, response time, buffering, backlog, priorities, and it would be stochastic. And the key idea in all of this, the key approach, was that0 the telephone network could not solve this problem because it was used to continuous communications and data-- communications and voice. In the data world, you occasionally send out a burst, and then you're quiet for a long time, and the telephone network would use this technology, the application, very inefficiently. And I can explain why it was inefficient, but I recognized we have to take the communications resources and use them efficiently by sharing them, making them available on a dynamic basis. So when I send a burst of data and am quiet for a long time, you should be able to use those resources for your burst and a hundred other people should be able to interlace their bursts of excitement and long periods of silence, which was the case for data communications at the time.

Weber: Did people look at all at telegraph-telex-type networks, which are a little less-- they're not inherently as circuit-oriented as voice, right?

Kleinrock: So the existing network that was most like what I was conceiving was the telegraph network, because it was sending in bursts, was sending digital data, and it wasn't a continuous communications. The trouble is, there was no analysis of how it works. There's no optimization. There was no architecture. It was very simply, you know, bits going down a line. And the way the technology was working in those days is you had a link between two telegraph centers. It comes in from one to the other. When it hits here, they punch out a paper tape. They tear off the tape, they walk it across the room, they say, "Well, where should I send this?" "Send it to Boston next." Reads the tape and says... So it's very crude.

Weber: But the routing is all humans.

Kleinrock: It was all humans, and, you know, they had some aids that said, "You know, I know who is busy right now." So the routing was not there, the error-correction was not there, the evaluation, the analysis was not there, and the idea of understanding the principles was not there. What is there that drives us to work well here and not there? You know, one of the things I very rapidly found in my own research was that bigger is better. More traffic and more capacity is a big win, not only for the economics, but for the response time, for the efficiency, for the throughput, and I could prove that mathematically.

Weber: And why is that? Why is it not possible to explain?

Kleinrock: No, I can, but let me back off as to why queueing theory was important. A queue is the perfect mechanism for sharing a resource. If I'm the server and data comes in, or people come in that want to be served, I'll serve whoever is in the line, in some order, maybe last comes first or first-- but I'll serve what's there. I won't sit by idle waiting for something that hasn't yet arrived. The telephone company did that. If there was a connection being used and that connection, conversation, went silent, the connection was still devoted to that now-silent conversation. For voice, that works, because the inefficiency is not so great. For data, it's far too inefficient.

So you can't let a communication resource be dedicated to a particular communication link. It should be for many conversations, many sessions. A queue allows that. It says, "Here's data. I'll send it. Here's data. I'll send it." So it's very good at resource-sharing, dynamic access. You get it when you need it and you don't when you don't. So, why was this scaling effect working so well? Well, first of all, queueing theorists, who had been around since 1917 when Erlang started queueing theory, they had no idea that this scaling would work, because they didn't believe a server couldn't work any faster than one second per second. A human clerk can do one second of work per second.

A data line, a communication line, you can have 100 bits per second or a megabit per second, so you can scale up the speed of the server. And basically what happens is, if you look at the equations, the load… if I increase the load, the demand, and the capability of a server, say, a communications line, I can keep the load the same. It's going to be busy 40 percent of the time. But each piece that comes through gets served that much faster, and so the queue empties faster, and each transmission.

Suppose there was no queue. Suppose there was just someone-- a piece of data came in that needed to be transmitted. If you scale up the capacity, it's going to go through faster. I mean, it's obvious. It works even when you're waiting online because everybody's going through faster, so the entire queue empties more quickly. Really I need to show you the mathematics, but it was a revelation that bigger was better, and big time. You can drop the response time, you can increase the throughput and increase the efficiency, all at the same time. I mean, what else could you ask for? So that was a consequence of the model I set up.

The challenge of my own work was to set up a model of data communications, think about what the routing procedure would be, think about what the topology should be, think about how large the capacity should be in each line, and then optimize all of that. So the first thing I did was to send in a thesis proposal where I decided: what's my approach and what previous work I'm going to use? So there was some interesting things going on at the time. I handed in my proposal in 1961. The late 1950s was the era of timesharing.

Weber: Yeah, I mean, tell me if I'm wrong. You were mostly thinking in terms of real-time interaction. I mean, theoretically, you could also network back to process machines, but you were thinking of...

Kleinrock: I was thinking of computers talking to each other, file transfers, as well as people talking to remote machines, sending communications back and forth. In that case, it's more interactive. In the file transfer, it's more steady. But the fact that timesharing was around, and I was deeply involved in that from my earlier experiences in my work, the thing that makes timesharing work is the same thing that makes data networks work, the following... You have a computer-processing machine. You can use it a few different ways. One way is you can sign up for use of that machine for an hour at a time. That's the way they used to use these machines. You sit there at the console, you write a program, you run it, it doesn't work, sit there scratching your head, and you say, "What's wrong now?" So the machine is dead idle while you're trying to figure out what's wrong. You try it again, it doesn't work. So what they did in the early days is they said, as machines got more expensive, we can't afford to let someone have that luxury, so they threw the user out of the computer room, and they said, "You want to do a job, you submit it to an operator and we'll batch it along with other jobs and we'll pipeline it through."

Weber: It becomes a queue, it looks like.

Kleinrock: It becomes a queue, but in this case, you don't get the big advantage. You get the efficiency of the machine, but the response time is terrible. You submit your job, two days later you get the printout that says, "Syntax Error, Line 44." You say, "Oh, my gosh." Feed it again, two days later, another syntax error. And finally, you get a logic error. You know, your thinking was wrong. Weeks and weeks to do a program. Finally, they decided to let the user back into the computer room, and they said, "Now we're going to give you remote access, and we're going to let many people use the machine at the same time. Your job runs, when you're done, the next guy runs his job." The next guy runs his job, and he sees his results immediately and so do you. So the idea of taking a resource, in that case, a processing machine, and sharings it among bursty users in a dynamic fashion was exactly the thought that I brought to communications links. These links, don't dedicate them. Share them dynamically, resource sharing. And that was the big idea, and that was an important idea. Now, the manifestation these days is in the form of packet-switching networks. That's only one way to implement that fundamental idea. So I started to do that. I developed the model, and the mathematical model that I needed -- recognized that I wanted to do the sharing in a queueing network -- was there had been some work in 1957 by a fellow named James R. Jackson, where he created a job-shop model where jobs moved around in a job shop to get processed here, something is cut here, threaded here, screwed here, welded there, from station to station. And he made a model of random jobs moving around in a random environment, and this model was very appropriate for my computer network. Data moves around, gets processed along the way, hop, hop, hop. The trouble is his model assumed a kind of a... no direct path. It was random path that he talked about instead of a known path through this job shop. So I took his model, I modified it, took the ideas of resource-sharing--

Weber: And what made you think of this model as being appropriate?

Kleinrock: Well, I recognized that with this dynamic resource-sharing that a queue is the right solution, so I started studying the queueing literature.

Weber: Okay, in all sorts of fields.

Kleinrock: In all sorts of fields, but his work on networks of queues was almost the first work on networks. I knew it had to go hop, hop, hop through a network, but what happens when you go from one hop to another? What's the total response time when you go down a path? He had thought about that problem, so I took his model and adapted it. Wrote up a proposal saying we are going to take a case where we have nodes that store, sort, correct, route. The thing is called data messages. We're going to use the model for how the performance works, and I'm going to look for all the obvious things, the throughput, the response time, the buffering, and all the rest, the correctness, the routing. Submitted this proposal, and it was accepted in... I believe it was June of '61. I submitted it in May of '61, or maybe it was April. I think it was May '61.

So I started working on this, and I recognized very soon that when these messages arrive at nodes, there's competition in that if there's a queue, who gets served next? And I recognized if you could change the order in which they get served, they might affect the overall performance, because it was pretty obvious, if you had a long message-- it could block short ones. Let the little guy through. It doesn't hurt the big one much and helps the little guy a lot. So I started thinking about priority queueing, and I recognized that you don't always know which the big and short messages are necessarily. So if you give a message a short amount of service, and if they don't finish, stop it, it's a long message, get somebody else in. So they either give it service in fixed intervals, and if the guy fails, send them back in, get somebody else in front.

So the idea of taking the message and chopping it up into fixed pieces, into bits of work, and feeding them-- A short message is just going to have a little bit of work, maybe two packets and it's done. Long message, it's a hundred packets and it's done. We didn't use the word "packets" at the time. So I said, "Let's consider what that would do to the performance of a single node," if you had this -- I called it "round robin." And I was able to evaluate it exactly, mathematically, and show that this was great in reducing the problem that we recognized. Long messages in front of short ones is a real problem. And I showed exactly what the performance was as a function of the message length and the packet length and all of that. So I published that paper in a report at MIT Lincoln Labs. In fact-- no, it was a Research Laboratory for Electronics (RLE) [report] at MIT, and I described exactly what was going on. And that was the first paper analyzing the idea of chopping things up into little bits and analyzing it.

Weber: But in this case, the little bit is the amount of processing time.

Kleinrock: Exactly, but I applied that, I took the concept of needing it in the time-sharing world, and I said, "What would it do in the data-networking world," because I specifically said, "What will be priority queueing nature? How would that affect the performance of the message going through?" And I applied it to the networking application directly. Okay, so, I did that, and I worked on my dissertation. I was able find principles, the resource-sharing principles, the bigger-is-better principle. For the routing, I introduced the idea of distributed routing. And there's another impact from Shannon. Shannon's main work -- you said law-of-large-numbers kinds of things before. The fact that we needed to statistically take advantage of certain ideas, Shannon used that exquisitely, and in his coding work, the longer the code word, the better it is, the more it's predictable, for reasons based on the law of large numbers.

So he was always interested in large sequences, large effects, the effects of large numbers. So when I decided to do this research on networks, I said, "I want to study this for large networks, because I know when you enlarge the scope, properties emerge, behavior emerges that you won't see when you have one, two, five, ten nodes." So my thesis proposal was called "Information Flow in Large Communication Networks." The final title was not that, but it influenced my thinking. I wanted to copy Shannon's approach. What will large mean? So I said, "If we're going to have a large number of nodes in this data network, you can't put all the control in one place. It doesn't scale. It's too vulnerable. It's too much load. There's too much communication traffic to get all the information in one place."

And so I decided to distribute the control. That's the birth of an event. It may wait around and finally get a free line. The call takes place, and then it's finished. It dies. So if you look at the number of people making calls at any one time to a certain number, a new birth, a new call, the finish of a call, a death, and you want to see how big is that population. Birth, death, and if you know how big that population is statistically, you decide how many lines to make available so there's enough for that size population.

So you see where the mathematics is leading to your good engineering solution. And now that's a very finely developed theory in mathematics and stochastic processes, and now in queueing theory. And I used the same technology and I developed some more, because that model is such a beautiful model for solving all kinds of critical problems. Anyway, that's Erlang.

Weber: So you were reading then, as it were?

Kleinrock: Well, when I needed to find out about queueing theory, I looked around at the books that were available, and the best book I found was by Philip Morse. He had a book called "Queues, Inventories, and Maintenance." And Phil Morse was a great, great mathematician. Morse and Feshbach, that's the Morse of Morse and Feshbach. If you don't recognize it, that's okay. But his book was sort of a strange view toward queueing theory. It was his particular view, and it wasn't sort of an introduction and develop to more advanced stuff. So I started looking around some more, and then I dove back in the past, and I found Erlang's work, which was now more formal, and I found some of the very mathematical books by Pakatch [ph?], by Kohn [ph?], by Propp [ph?], some of the more theoretical books.

So I taught myself queueing theory, and later on, I started teaching the course and developed my own book, which took it out of the mathematical realm alone and brought it into the engineering/mathematical realm, because I had to talk to engineers who were going to use this, had to learn it and then use it. And so when I wrote my books in the mid-'70s, it was with an eye toward making it accessible to electrical engineers who were very good in mathematics but needed to see the way in which the theory tied to the problems they were trying to solve.

Weber: So you had mentioned that you were certainly around timesharing a lot in that period. Describe your involvement.

Kleinrock: So, timesharing was the rage in the '50s, when I became a graduate student. Computers were being built all over the place, and at Lincoln Lab, there was a particular group. It was Group 22. The head of it was a fellow named Wesley Clark, and he was developing the first transistorized computers, TX-0, TX-2, and those are the machines -- in particular, TX-2 -- on which I did my simulations and learned all of my computer technology. That group was developing timesharing systems to run on the machines they were building. And timesharing, which was developing all over the place…Time-sharing, in the late '50s, was being developed in a number of places, and people were proposing timesharing algorithms, who gets served next and for how long, so people could share the use of the computer. And at Lincoln Laboratory, we had the TX-2 machine, and that was another very elegant time-sharing system. And we were constantly trying to find ways to introduce new algorithms to get better performance for the users, but there was almost no analysis as to how these new algorithms would perform.

And I recognized that timesharing was a queue of people waiting to get served. It was a priority queueing system, who gets served next, and it was a highly preemptive priority queueing system, which means when someone's being served, you can throw them out of service, put them back in the line, and serve somebody else if there's a reason to do so. So I recognized that's a really fascinating set of queueing theoretic problems. So I jumped in, and this is after I finished my dissertation, and recognized all the queueing-theory knowledge I had and expertise could be used to analyze not only computer-network systems, it could be used to invent and analyze priority queueing systems and timesharing systems, because both of those concepts were being used in the data networks. How are you going to serve the messages? So I was able to satisfy the timesharing community with results that they could use immediately, and developed techniques which I could then use in data networks, none of which existed at the time. See, the dilemma was, when I finished my dissertation in 1962, I handed in my dissertation, nobody cared. Nobody was building data networks. There was no data to send. Machines weren't talking to each other. So I kept developing the theory and improving it and finding applications in the timesharing world, but these were directly transferrable to the data-network world because we needed a kind of priority-queueing system in every node to improve the performance of the traffic that was being served by these various nodes.

Weber: And you did your thesis with-- You had some illustrious classmates, I believe. Talk a bit about that.

Kleinrock: So at MIT-- MIT, I'm happy to say, I was there during a golden era at MIT. I mean, Shannon, Minsky, McCarthy, Huffman, Rosenkraft [ph?], Wiesner, Elias, Fano. I mean, these guys were giants. And happily, the graduate-student population were also a bunch of brilliant people. My classmates, who would remain my colleagues and friends through my entire professional career, were names that are well-known to this oral history. Larry Roberts was a classmate of mine. Ivan Sutherland was a classmate of mine. Who else would you recognize? Tom Stockham was a professor when I was there. Irwin Jacobs was a student of Ed Arthurs. Irwin was his first student. I was his last student. He graduated five students. Irwin became a professor while I was at MIT, but he was a graduate student first.

So the people there were just enormously capable, doing magnificent work, and the thing about MIT-- Tom Keiliff [ph?], for example, Jack Ziv, these people more in the information-theory and communication-theory side. The thing about the MIT environment is that it was one of collegial interaction. The world of the graduate student, we were put in these bullpens, many desks in one room, and we all shared our ideas. And when you come up with something, you test it against your classmates, and they would critique it and help you sharpen it. And whenever you got a result, you would announce it to your classmates first. The idea of getting a result was kind of a, you know, you ring the bell, "Look, I just proved the following," "I just developed this," "Such and so acts like so and so."

So the environment was good. I remember Jack Ziv and Tom Keiliff [ph?], from very different worlds, working on very different problems, they were going to each other's cubicle, and they had a schedule of teaching each other what they knew. Jack would teach Tom, Tom would teach Jack, and it went on for a semester or two. I mean, the kind of intellectual attitude and enthusiasm was overwhelming, and all ships rose with that tide. It was really magnificent. And so, you know, I would discuss my results with, particularly, Larry Roberts and Ivan Sutherland, because Larry was also a staff associate at Lincoln Lab getting his PhD. See, when they urged me to go on for my PhD., they said, "We'll do it with another staff associate program," so they paid for everything in the same way. It was a very, very liberal, liberal approach by MIT and Lincoln Lab. And the work I was doing in the summers was fantastic. That's where I met some of these people, Ken Olsen, Wes Clark, Jack Raferel [ph?], Frank Heart, on and on.

Weber: And you were doing things for Ken Olsen throughout this period?

Kleinrock: No, no. Ken Olsen, only 1957. He left. Then I started working for Jack Raferel [ph?] on thin magnetic films. That's where I did the Kerr magneto-optic effect. And then when I started working on my data networks, I was working there in Lincoln Lab on those problems, because they encouraged research and they encouraged interaction. So the environment was very strong, and the kinds of things that I was able to do were just gratifying. I remember when I had to make my thesis defense, it consisted of two parts, an oral part where you described the analytic and the optimization and the mathematics, and then a demo, because I had written a very extensive simulation program to prove out my ideas. It's not enough to just put it on paper. You have to show it works. And so what I did was I wrote this long simulation where I ran a system, a data network, without the assumptions that I needed for my mathematics. Then I compared my mathematics predictions with the actual results of the simulation, a very realistic simulation, and they were almost right on. So it showed that the assumptions were very good.

Well, I had to present this as part of my demo of my thesis defense. Larry Roberts did the same thing. He had basically a three-dimensional hidden-line program he developed. Ivan had developed Sketchpad. So the three of us gave our demo part of our thesis defense simultaneously to the union of our three committees, so it was Shannon and Minsky and Arthurs and some really wonderful, about seven professors there. And we did very well. Ivan did his Sketchpad thing, and one of the things he used Sketchpad for was to build a truss for a bridge, and he would then put constraints on the members of that truss which were inconsistent, because one of the things that Ivan did was he was able to recognize if you can't always get exact solutions, you'd look for the best approximate solution. So he put in inconsistent constraints and the truss would have to adjust itself to end up with the least stress and strain on each member, still satisfying the constraints of being so long, in this position, or not exactly that position, etc. So, in short, a very nice demo. And then, while he was facing away from the computer talking to the committee, Minsky wandered around to the computer and he created his own truss problem. He put in constraints and he said, "Satisfy." And this thing started trying to satisfy it, and it was an unstable situation and the elements were exploding on the screen and ballooning out, and Ivan didn't see this. This craziness was going on behind him, and we were looking in horror at what was going on. <Laughter> Finally, Ivan turned around and laughed along with the rest of us. But that's the kind of individuals I was privileged to work with. It was a magnificent group.

Weber: And so when you all gave your demos, it was literally back-to-back in the same room?

Kleinrock: Same, back-to-back, exactly.

Weber: And with a graphics terminal for...

Kleinrock: Yeah, it was the TX-2. It had a wonderful screen. We had a light pen, so it was interactive and graphically very good. And, you know, a lot of that technology was developed by some of the people in the room right there. See, Larry developed something called the Lincoln Wand. That's still not available these days. I think with [Microsoft] Kinect it's the first time you see that. He took a long baton. He put a microphone at the end with a wire into the computer. He got four spark plugs, one on each corner of the display, and they cycled around -- blip, blip, blip, blip -- rather quickly, and this microphone would pick up the blips. And looking at the time delay from each, you could tell where the wand was in 3-D space. So you could point, have these virtual points appear just the way we do now with simulating people in today's high technology. The light pen was developed there.

Weber: Sorry, you gave me, actually, a film of the Lincoln Wand.

Kleinrock: Yes.

Weber: But, I mean, this was primarily as a kind of three-dimensional mouse...

Kleinrock: Input device.

Weber: ... you could use for virtual-reality sort of.

Kleinrock: Absolutely, but there was no virtual reality used then. Simply, it was a pointer, and, in fact, it wasn't used very much. It was more a demo. The light pen was used by everybody, and that was a very useful device. I mean, I used it. You had all these control buttons. The user interface was terrific. At any rate, the environment in which you find yourself is as important as the contribution you make, because it makes that contribution all the more effective and dramatic in the sense of it lifts you up. Like with the Eagle Scouth, it sets a target up there which you just have to reach a little harder to get, and you go after it.

So, graduated from MIT, ready to go to Lincoln Lab, be a researcher. And Lincoln Lab, in their wisdom, said, "Before you come here to work full-time, look around at what's available out there in the world. I mean, don't limit your horizons." And so they encouraged me to, and I did take a tour around the country at various research labs to see what there was. I came out to California to Hughes Malibu Research Lab at the time. I checked with some of the other defense-industry research labs. And since I was coming out, I was encouraged to visit Berkeley, the University of California up north, and just talk to them about a possible position there. I was not interested in teaching. I wanted to be a researcher, although I had done some teaching at MIT, and not just as a teaching assistant. I took over a whole class for a semester, and I found I loved it. But it never occurred to me to even-- Look, it never occurred to me to get a Ph.D. Never occurred to me to become a professor. It was not in my vision.

So I went to Berkeley and I gave a lecture there, and one of the people who interviewed me was Lofti Zadeh. He was one of the big people in linear systems. And I left. At that time, the chairman left his position as chairman. The new chairman came in, and they lost my file. So over the next month or two, I heard nothing. Suddenly, and this was the spring semester of 1963, when I was basically spending a lot of time with Lincoln Lab and some time at MIT, and who comes for a sabbatical? Lofti Zadeh. And he meets me in the hall, and he says, "Kleinrock!" He was not as friendly as that when I was interviewing. And he says, "Hello, hello," and he thought that I wanted a job on faculty.

So he contacted one of his good friends here at UCLA, Balakrishnan, and Balakrishnan invited me out here to interview for a job. And I did, and I quickly got an offer, and now here I am sitting with an offer as an assistant professor all the way across the country, 2,500 miles -- in those days, that was like out West -- at half the salary I was getting at Lincoln Lab to a job I didn't really understand, and so I spoke to my people at Lincoln Lab and I said, "Look, I have this offer. It's appealing in some ways, but I owe an allegiance to you. What should I do?" And in their great magnificence, they said, "Listen, try it. If you don't like it, come back."

Weber: What's the link with the lost file at Berkeley?

Kleinrock: Oh, Zadeh thought I wanted a faculty position. I didn't. So Balakrishnan suggested I interview. I said, "All right, I will."

Weber: Oh, okay.

Kleinrock: So it was an accidental interview, in a way. It was not something I really wanted. So I came here, and I'm still here. <Laughter>

Weber: Well, you tried it.

Kleinrock: Yeah. And I have to admire Lincoln Lab's attitude, the people that I worked with, and that environment at MIT, which was so collegial, and somehow you grow in that kind of an environment. And I want to make that point that culture matters very strongly, because it's going to come back later when we talk about the ARPANET development. It's key. It's sort of, again, another golden era which I was privileged to partake in.

Weber: So you came out here...

Kleinrock: Okay, so, I came out in August of '63 with my… I had two children by this time. Drove across the country, started working at UCLA, started teaching information theory, and some minimal networking theory, but that wasn't all the vogue at the time. Nobody in the industry was interested in data networks, and in particular, the largest network company in the world, AT&T, and I spoke to them quite a bit at conferences and saying, "Look, here's a technology you ought to adopt because data networking is going to happen." And they said, "No, never going to work, and even if it does work, we want nothing to do with it." And I remember having discussions in public with those telephone companies at the Spring Joint Computer Conference and the Fall Joint Computer Conference. Long before the ARPANet was conceived, there would be these panels.

On one side were the telephone guys, in public, a ten thousand person audience. The other side us data guys, the computer scientists. And we'd say to the telephone guys, please give us a good data network so we can move data around. And they would say, what are you talking about? The United States is a copper mine. It's got telephone wires everywhere, copper wires. Use that. We said no, no you don't understand. It takes you 35 seconds to dial up a call. You charge me a minimum of three minutes of talking time, and I want to send a hundred milliseconds of data. And they said little boy go away.

Now, interesting comment, little boy went away and a bunch of us ate their lunch, eventually. But they were right for the short term. There was no revenue in data. There was no data. They were making huge amounts of money on voice communications. And the network was serving the voice very well.

Weber: They were also connecting time sharing to some extent and profiting.

Kleinrock: And but that was not a big revenue for them.

Weber: I mean they were probably looking at that as well we are connecting users to terminals--

Kleinrock: So, we know how to do it, but that was not the nature-- there was not sharing. When I'm connecting to the machine that's just me going to the machine, no one else is going to share that line. But deep in the network you got millions of people sharing. You're quite right about that. So, I can continue to do my work of publishing, developing students because there are wonderful problems left in this area. And so I started churning out a small army of networking experts until interesting things happened with in networking.

Weber: Step back just a little bit. So, you came out here and you were hired as a professor of--

Kleinrock: An assistant professor, seventy-five hundred dollars for nine months, and all the consulting you could do on one day a week. And that mattered because I did consulting to supplement my income. And I did take summer jobs, as well as a consultant or whatever.

Weber: So what sorts of things?

Kleinrock: Well, I remember I worked for a company called Babcock Electronics down in Costa Mesa. It was, again, a kind of military application to provide communications for the military in the field. And I introduced some of these ideas of data networks. And they were very well accepted, but the implementations just didn't happen in that time frame. But being a professor I found I loved. I remember one of the first classes I taught was an extension class, which meant you could take it day or evening. And so, it occurred in the evening. And so, a lot of the people coming in were from industry. They could get credit for it. And so, most of them were older than I was. And I'm teaching this-- little squirt with the crew cut, young, and these guys-- these experienced guys out there. And I'm in the first couple classes, they start throwing these questions at me. And they were challenging questions. And I realized they were really trying to squash me down.

Once I realized that, this question came at me at this level. I answered it at this level. And I went bang, bang, bang. And I nailed that guy into the ground. And then there was never any question. <laughs> You know that was the test. But I remember it suddenly clicked, here's what's going on. And I had to open up the full artillery. Yeah. And it worked. But I loved teaching then. I love it now. And it's a remarkable profession. I mean I could list a dozen things that make it the world's best profession-- if you like teaching. Unfortunately, a lot of faculty don't like teaching. So, it's misery for them. But I enjoy teaching. You work with brilliant students, undergraduate and graduate. That keeps you young. You get to do the research on anything you want. It's your choice. You don't really have a boss. You're independent. You can put extra effort in through consulting or other kinds of jobs and get more gratification, be it financial or challenging. You get to travel. The best people in the world come through the university to lecture. You're independent. That's a very important quality. It's a prestigious job. You're doing good. It's gratifying in so many dimensions. It just goes on an on. And it does keep you young.

There's no question about it. Just the brilliant young minds coming through, and the best way to learn is a subject is to teach it. So, I just love teaching. And so, that's the reason I stayed. And but the-- so the environment was terrific, except nobody cared about networking. And that was surely disappointing because I kept working at developing the technology. What's interesting is, that in parallel to what I was doing, Paul Baran was working on a similar technology with similar kinds of problems over at Rand Corporation. He was mostly concerned with vulnerability of networks. And he built in redundancy into the links. And then he began to think about it some more.

The chronology is that in April of '62, I published that paper on chopping messages into packets and serving them independently-- fixed length packets. In September of '62, Paul, who'd been working on this, published a paper talking about packetization. Nobody used the word packets at the time, just message blocks. And some routing issues and some other issues like-- we had no idea of each other's work at the time. It was totally independent.

In 1964, he published his very large set of reports on this. In 1964, I published my PhD dissertation. It was the first book on networking. It was called "Communication Nets: Stochastic Message Flow and Delay." And so that work was going on. But Paul experienced a similar thing. He tried to get AT&T to adopt his technology. He got the same stiff arm. They then tried to take it to one of the branches of government. And they said no. At that point Paul stopped beating his head against the wall trying to get it adopted, and went off to do other things. I stuck with the networking because it was an object that was, in my mind, [of] great value and a great challenge analytically and theoretically and technologically.

Weber: That's very good. But you were talking about taking a message and chopping that into processing, but the message could be any length when it came in.

Kleinrock: Yeah, when it came in it could be any length. Chop it up into fixed length packets to move it through. I considered many things. I considered keeping the message in its entirety and using priority queuing, which wanted the messages in totality get served, in what order, also breaking them up into smaller pieces. And I examined all of those analytically and there are a number of those in simulation, as well.

Paul chopped them up also into message blocks, fixed length blocks. And he was more interested in the architecture and the engineering aspects. I was interested more in the mathematical theory and some of the principles that emerge out of doing this kind of technology. And so, I was able to bubble up some of the issues, the principles of what was taking place, some discovered, some developed. And Paul's eleven volume series was very valuable. But they didn't play a direct role in the development of the ARPANET until much later. And Davies, Donald Davies, the other participant here, started thinking about these things in 1965, coined the word ‘packet’ and started… published the first essential paper in ’67, tried to get a node developed. It was the Gatlinburg conference.

Weber: Didn’t he publish something in ’66?

Kleinrock: He published something in ’66 that was not widely circulated, at all. In fact, I’ve got a reference if you want. But the reason I’m bringing up Davies is he also met with non-support in his own institution. They let him build one node, instead of making a multi-node. So, one node does not a network make. And he also-- I don’t know this for a fact, but I imagine he was frustrated with the fact that his ideas and the later whole plan with some of his other team members at the National Physical Laboratory-- had NPL, the National Physical Laboratory in Teddington, taken his ideas and built the network on the timescale he was hoping for then the U.K. would have been the birthplace of the Internet instead of the United States. It’s an interesting historical quirk, which I think is important for people to understand. And there’s a bigger picture here, as well. Were these ideas brand new? No. They were in the air at the time I did my work, Paul did his work, Donald did his work, and ARPA decided to make a network. It was inevitable. And if you ask about where did the vision come from, I’d have to go back-- well, you can go back to logic and mathematics as being-- thinking about digital objects, but the first document that I’m aware of which really talks about this is Nikola Tesla. In 1908, there’s a quote where he talks about being able to send pictures, text, voice over intercontinental distances instantly using a device no bigger than his watch. And he goes on to describe what sounds very much like the Internet today.

Weber: Yeah, a World System, I think.

Kleinrock: A World System. And it didn’t include video because there was no video at the time. He talks about pictures, charts, text, voice. And then we have H. G. Wells. We have Vannevar Bush. And then we have the beginnings of the technology, which emerged into today’s Internet. So, I mention this because if you ask where does the credit belong for this Internet, the answer is everywhere. It was coming. People were thinking about it. There were a few high points, but it’s a shared credit, a shared development.

Weber: For the idea of-- I guess it would be packets or breaking things up, how much did internal computer architecture affect thinking? In other words, within computers, you’re to some extent dealing with some of the same issues that you deal with between computers.

Kleinrock: In the sense that there are small modules of things moving around, instructions in the case of a processing machine. I don’t think so. In my mind, the thing that attracted me was the way in which the job was served in timesharing was similar to the way the message should be served in networking, rather than the internal way in which a processor looks at microinstructions and tears them apart. The-- everything has it’s time. And as technology develops and it engenders certain types of applications which then drive the technology again. The technology was right at the time the Internet appeared. And if anyone is articulate in making that connection it's Larry Roberts. He wrote this magnificent paper. I believe the title was "Data by the Packet" where he talks about why did the Internet begin in 1969. And his conclusion is that the cost of communications and the cost of computing were following a trajectory. The cost of communications was lower than the cost of processing in the early days. So you could waste communications to save on processing. What kind of processing? Routing, packetizing, error control, flow control, just throw horsepower, basically, processing at it, and you'll-- communications at it, and you save the processing.

Weber: It's irrelevant because the communication is cheap.

Kleinrock: But as the cost of switching came down, you could now do enormous amounts of switching and processing to save the long-lines [long distance] communications cost. Now, Larry happens to target the crossover at 1969. And a lot of it depends upon his assumptions. He probably could have jiggered it to come out '68 or 1970, or a year-- doesn't matter. It was about the right time. And sure enough the economics and the technology supported the introduction of a real network using this kind of technology.

Weber: And Larry Roberts was doing-- I mean how much were you exposed to his early networking experiments?

Kleinrock: Okay, so the relation to Larry Roberts is a very interesting one. As I told you, I did my simulation on TX-2 at Lincoln Lab. Larry and I were both staff associates at Lincoln Lab. So we were at MIT and Lincoln Lab in summers. We were both using TX-2. One of Larry's first jobs was to write a compiler-- a meta-compiler for TX-2. He and I shared an office at Lincoln Lab. And everything I was doing on my dissertation, I was bouncing at Larry. And he was bouncing some of his 3D ideas and his compiler ideas at me. So he was totally familiar with all of my results. He even suggested before I got my dissertation that what I was doing could be applicable for optimizing the flow of automobile traffic on the road network. We never implemented that, but-- so there's that kind of an exchange. He scared the hell out of me one night. I was running my simulation, actually debugging the simulation.

It took months to debug this very large program I wrote. And I would get, before timesharing was used at TX-2, I would sign up for seven hours at a time from midnight to 7 am, four nights a week on the TX-2. I would get the machine. I would sit there. Spent a lot of time scratching my head, what's wrong with my program? Why this bug that bug? And that machine, the TX-2, was an experimental machine. And it had-- the console had pieces that were being changed, improved, repaired. And there were usually pieces missing. And I was there one night, late, at four in the morning, really tired, hungry, dry mouth, bearded. And I have this million dollar machine I'm running. And I've got to make sure it doesn't break. So you get familiar with all the sounds, all the lights. You've got to make sure the machine is healthy. And then I heard this strange sound. I heard <makes whispering sound>. I said oh what's that? That's very strange. Is something losing air? Is there something breaking?

So I started looking around. I looked at the console. Up in the console was a-- there was a piece about the size of a six inch ruler, which should have been holding one of the registers. It was being repaired. I looked up there. And I saw a pair of eyes looking at me. I was there in the room alone. Scared the hell out of me. It was Larry Roberts scaring me, going <makes whispering sound>. Like that. <laughs> Could have killed him. We had a good laugh about it. But those shifts were pretty hard, by the way. Four days a week, but they were not contiguous days. So my sleep patterns were outrageously disrupted. Even in session, didn't help when I was back as an undergraduate.

So even to this day, my sleep is not predictable. I mean I can fall asleep any time I want for five minutes or an hour at a time. I go to sleep very late at night. I won't necessarily wake up early, but-- my wife hates it because she's a regular person. She likes to go to bed early, wake up early. But the nice thing is I can adapt. I never get jet lag because I'm in some sense always jet lagged. Anyway, that was the story with Larry.

Weber: Describe when you first met Larry Roberts.

Kleinrock: Okay, I met him at Lincoln Lab when he--

Weber: What was he like?

Kleinrock: Oh, Larry was a-- he got his undergraduate at MIT, also. He was one of these hackers. He would play all kinds of tricks on the administration, on his fellow students, on his girlfriend, using high technology. He was great at this. And he got slammed down a few times, but he was a real tinkerer. And he loved pranks. But he's one of the smartest guys I know, really brilliant guy. He would get things quickly. Somewhat impatient, not very-- what shall I say, congenial in his discussions. He calls a spade a spade. And he doesn't tolerate fools very easily. He's still like that in many ways. But that's part of his strength. He thinks through things very quickly. And he goes into enormous detail.

When I shared that office with him, he was writing his meta-compiler. I was happy if I could write a program in this language. He wanted to know what that bit did in the word when an error occurred of the following type, was it a one or a zero? And how could I use that in my compiler. The level of detail I never wanted to get into. He was that kind of detail person. And it did him well. He wrote a great compiler. When he does a design, even today in some of his networking ideas, he's right there in with the engineers doing it, deep, deep knowledge.

Ivan Sutherland, another person, is worth talking about. At MIT to-- in those days, to get your doctorate, you had to pass a really tough exam called a qualifying exam. That really separated the men from the boys. And you could fail it once, the second time up or out. And half the people failed it each time. So you basically didn't have a very good chance of passing through that filter. And you take it between the

Master's and the Ph.D. And it's based on fundamentals, but it's all tricky questions in those times. Nowadays, it's more standard stuff. I mean if you didn't see the trick, you couldn't get the answer.

Now, Ivan got his Master's degree at Caltech. He came to us after his Master's. So within two months, he had to take the qualifying exam. The rest of us got our Master's at MIT, so we knew the material that presumably these trick questions were about. Ivan takes the exam, number one. He did better than anyone else. I mean he just leaking intelligence, brilliance, flash. He's very fast. And, but his personality also was totally asocial. He wouldn't say hello. He'd just start talking. I mean he called me on the phone because we had a lot of conversations, and he's start with describing a problem he had.

I'd say Ivan, just a minute. Say hello. Take a breath. And slowly, he began to get a little more social, but both of these guys have that characteristic of no nonsense. Let's get right to it, stick to it. And Ivan, also, call a spade a spade. You don't know what you're talking about. I heard him say that to many people. False. That was a favorite. False. Not, I don't agree. False. You know it's like a dictum from above. And he was right, usually.

Weber: Did he play pranks like Larry Roberts?

Kleinrock: No, he was not a prankster. No, Larry was a great prankster.

Weber: What were some of his other pranks?

Kleinrock: Well, I don't want to tell stories out of school, but he would manage to break into one of the safes of one of the administrators. I think-- oh, he got into his office, which was locked, got in and turned some things topsy-turvy inside some secret compartments. He did things to his girlfriend. He set up a-- every time she picked up the phone to call somebody, usually another boyfriend, Larry's voice would appear. I mean that kind of thing. And he drove her crazy. But Larry had some other interesting side lights, some of which I participated in.

We-- in 1968, Larry pointed out that a silver quarter was worth more than 25 cents if you could melt it because the price of silver was going up. But it was illegal to melt. But it was considerably higher. So we decided to collect silver quarters and silver dimes and wait until a day when you could melt it. Okay, so how do you get-- in 1964 is the last time they were minting silver quarters. From then on they minted those hybrids with a lot of copper in it. But there were plenty of silver coins in circulation. So what do you do? You have to get mixed bags of quarters and somehow find a way to separate out the silver from the other. So, I developed a machine that could do that. I built a machine that would handle 10 thousand dollars an hour. It was based on the fact that silver's a better conductor than copper, and if you pass it through a magnetic field, there's something called-- an eddy current gets developed, and it moves a little bit in the direction of the field. The silver moves a little more than the copper. So I ran these coins up to a coin machine, down a slot, passed them through a very heavy magnetic field, and took a bayonet blade from World War I. And with a little bit of difference, some popped that way, and some popped that way. Two laundry buckets collect them. And it worked just fine. In fact, to make it even better, do it twice with the good ones.

So you don't have any bad ones in the good one. Okay, ready to go. So now we need some money. We've got to get-- we want to buy these coins. Well, we didn't have any money. But one of my colleagues at UCLA had five thousand dollars. And he wanted to play into this game, also. So, Walter gave us five thousand dollars. Our plan was to go to the bank, buy five thousand dollars worth of coin, and pay for it. Take those mixed coins, separate out the good ones, give back the bad ones, get five thousand dollars worth of good ones, go to the bank, and say we want to borrow five thousand dollars. And for collateral, here's five thousand dollars, but put it in the safe, the silver coin. We'll pay you interest on the money you loan us. Okay, that was the idea.

So now we're ready to get that going. The trouble is the banks wouldn't give us any coin because the U.S. government was trying to do that same thing. And they had a ten-thousand dollar machine built out of Cincinnati Milling Machine Company to do this. My machine cost about a hundred and fifty dollars. So we couldn't get-- so we started going to the laundromat machines because they had a lot of quarters. And I ended up dealing with the mafia a little bit because they owned those machines. I couldn't get enough raw quarters. They were being held up. So we ended up getting some person from San Diego who would hire Mexican laborers to pick them out by hand. And we paid him a premium for that, as well. So how much did we collect. Well, I won't give the exact numbers. There were four of us now. There was Larry, myself, and two others, one of them the fellow who gave us money.

Those three decided they wanted to collect, let me call it, D dollars. I decided to do ten D. Ten D was such that I could just about afford to pay the interest in ten D that I'm going to borrow from the bank. And that was my wife's salary, Stella's salary at the time. So, we set up to do this. We collected the money, put it away, and then if you remember, the Hunt brothers tried to corner the market. The price went way up. The government slapped them down, took the price down well below where it was profitable for us anymore. And the other three panicked a little bit. The money-- the price came back slowly to where they could just about break even. They sold theirs. I sold enough to own the rest. And I still have it. I still have the silver coin, actually silver's doing pretty well these days. At any rate, that was one kind of thing that Larry was engaged in.

Second thing, Larry liked to play blackjack. And so do I. And Larry and I developed a system for counting cards around the same time that Ed Thorpe did. Ours happened to be a better system. He had the ten-count system. He also had an ultimate system. Ours was better than his ultimate. At any rate, we developed a system. And we went to Las Vegas a number of times using the system and did pretty well. And there's some fabulous stories I won't go into here, but that's another side of Larry, very, very eclectic. Once, he and I drove up to Bangor, Maine to see a full eclipse of the sun. There was a kind of camaraderie and a lot of interesting adventures that we had. I've got to tell one of the gambling stories, nothing to do with blackjack.

We also recognized that a roulette machine is a candidate for winning. If you can predict which half of the wheel the ball is going to fall in, you can get almost two to one odds in your advantage. You expect to lose a nickel on every dollar you play on roulette. So if you just play straight roulette, you're going to lose. And Shannon's law of large numbers approach says for sure you're going to lose. So if you must play roulette, take all of your money make one bet. If you win, go home. If you lose, you're gone. It's going to happen anyway. So the point is, don't let nature get you with a long run because she'll take you apart.

Anyway, we decided to try to measure what's going on with the roulette wheel. Now the wheel is running one way. The ball is going the other way. And it's Newtonian mechanics. The ball will fall off the raceway at a critical speed and bounce on a half a wheel. Sure they have those spoilers, which confuse you. But you can get an estimate as to which half of the wheel, if you know when the ball's going to fall.

So we developed sound. We know the wheel rotates at about a half a revolution per second. We need to know the speed the ball is going at. So, we need to make some measurements. So Larry took a microphone, wrapped it up in his hand like he had a broken arm, took the recorder-- they were big in those days, stuck it in his jacket pocket. And he wanted to get near the wheel so he could hear the ball go <makes whirring noise>. And get an idea of the dynamics and the kinetics of the ball. And I went down with him as a kind of decoy. So he goes. He puts his hand there. And he starts betting. And I start betting. And he's making measurements.

The trouble is I start winning. I start winning, not on black and red. I'm winning on the numbers. Pure accident. But guess what? That drew a little bit of attention to us. They see we walked in together. I'm winning. His hand is next to the wheel. So they take Larry with his broken arm. And they drag it. And they say, "Let me see that arm." You know he's supposed to have a broken arm. And at this point he says it's my broken arm. He pulled it back. And the two of us ran the hell out of there. But that's the kind of things we were doing.

Weber: Did you come up with a system for the--?

Kleinrock: Actually we did. We did. In those days, people wore tie clips. What you do is you-- when the ball crosses double zero, which is green, double zero, you hit it. You wait for three more passes on double zero, click it again. And that give you the data you need to say how much-- from that double zero, where the ball's going to fall off the wheel. And you know where the wheel is because you have a little calculator. So we put together a calculator. Damn thing burned up in his-- he was carrying it, in his pocket. You know it just smoked out, so we never did it. But the theory-- it was enough to realize we could do it. That was the important thing to us.

Weber: Some of the blackjack stories you have told in other venues haven't you?

Kleinrock: Yes. Yes.

Weber: I mean anything that's already out feel free to say.

Kleinrock: Well, this one doesn't involve Larry, actually. I was here living in Brentwood. And the word got out that I, and actually Walter Karplus got involved in this also, trying to measure roulette wheels because not only can you win by the Newtonian mechanics, if the wheel is not perfectly balanced, it will bias certain numbers. So Walter and I went downtown to measure wheels that-- the roulette wheels were being built here in L.A. for Las Vegas. And word got out that I'm sort of interested in stuff. So one day I get a call from a guy named Alan. I won't tell you his last name. And he says, I and my colleague this summer were basically dealers at blackjack up in Horizon, Lake Tahoe. And we'd like to talk to you a little bit. They came down. And we started talking. And pretty quickly I put aside the roulette stuff because they started telling me how they were dealing with blackjack. These guys had system also. And this guy, Alan, had more memory cells than either Larry or I.

And he was able not only to do the regular system, he could count how many threes and fours were left in the deck. And I'll tell you why that's important in a moment. He related the story that he had-- this summer he was going to spend it in Las Vegas trying to make a killing. That's why he wanted to know about the roulette. But on blac jack side, he was playing there for a while, downtown-- the scrummy part of town. And he was playing and somebody called him over and said look, son I see you're playing pretty well. I'd like you to play with my money because the kid didn't have any money that he could play with. He said whatever you win, you keep ten percent. Thirty percent I'll keep. And sixty percent, we'll give to the dealer. In other words, they were buying the dealer. And what can a dealer do for you? And by the way, the club owners hate this. They really want fair play because the Nevada Gaming Commission will come down on them. The dealer has to give you very little information to give you a huge advantage. And it's the following, when he shows a ten. He must peek to see if he has an ace. And show you that he has blackjack. Now the nature of that card is extremely valuable. And if he can just tip off to you either if it's a good card, a ten, a nine, an eight, or a lousy card, a three, four, five, or two, a six, just with some motion, some hint, it's enormously valuable. So the kid played. He couldn't beat the dealer. The dealer's relief came on. He beat the dealer for ten thousand dollars. They paid the first dealer off. The kid got his thousand bucks. So he's describing to me his ability to play so well.

So I said let's go to Vegas and play. He and I went there. We're downtown. This is a wonderful story. We're playing-- there's the two of us and the dealer. It's a hand held deck, which is dangerous because the dealer can do all kinds of manipulation, but it's good because there's not many cards left by the time you get to make your play. And if you know what's going on, you can tell what those cards are. And it was an open hand. So the kid gets a ten and seven. And the dealer offers to hit him. And the kids says hit me. Dealer says you don't want to hit seventeen. Kids says hit me. You don't want to hit seventeen. Hit me.

You really don't want to hit seventeen. The kid says, "Goddamit, give me the four," because he knew there were only fours left in the deck. <laughs>

Well, of course he won. A couple of plays later, they changed dealers. They moved in a mechanic, a card shark. And we started going down hill. We left the room. But, you know, because they can do anything that want. They literally are magicians. But the fact is the kid said, "Goddamit, give me the four." Stupid move. So when I play there-- when I was playing, you have to hide the fact they you're counting, obviously, and if someone sits next to you while you're madly counting and starts you in a conversation, you can't say, "Shh, I'm counting." You have to sort of parallel process. Or sometimes you have to make a dumb play. I remember one time I was playing downtown, head on with a dealer. And he said, "You count pretty well don't you, Sonny? I said, "Count, what do you mean?" He said, "You're counting the cards." I said, "No, I'm not. I can't even add them up." I said, "You mean some people actually count cards?" And his response was, "Yes, people will cheat in all kinds of ways." They considered then, and consider now, counting as cheating. They say it's a game of chance, not a game of skill.

So these are the dynamics you have to play with when you're-- it's not the narrow view. You need the worldly view that this transistor is heat sensitive. You've got to know more than just 'it amplifies.' So it's the worldly view.

Weber: So talk a little bit about the-- I think we're at the beginning of your involvement with ARPA. Oh no, sorry with Larry Roberts, how much were you aware of his early networking experimenting?

Kleinrock: Okay, so the first network experiment that he published was with Tom Merrill where he connected the TX-2 with an SDC machine. I read the paper. I wasn't interacting with him very closely then because he was at Lincoln Lab at the time. And I was at UCLA. But I read the paper. It was interesting. And I pointed out that it's great that you did that, and he bemoaned the fact how difficult it was, and how much the technology was needed to make that done properly. And we talked about the fact that he knew I had done that kind of work in my dissertation, and the shame that it hadn't been implemented by anyone at the time.

And the story ends there, until the ARPA connection comes in. But Larry was very much interested in networking from the knowledge that there was a technology, a latent technology, waiting to emerge, and no one was exploiting it. Larry was always recognizing the value of any new technology to new hard problems. And so when the ARPA story began, he got involved. We got much closer at that point because he wanted my involvement to help define the network, measure it, set up the first node, etc. So the ARPA story most people know.

ARPA was formed as a result of Sputnik in 1957. Caught us with our pants down. Dwight Eisenhower, our president, said that'll never happen again, decided to create this advanced research projects agency, which was going to bring back the prime position the United States had in science and technology. And so they started supporting research in a variety of sciences, including computer technology. And then they brought Licklider in in 1962 to form the information processing techniques office. Meanwhile,

Licklider had already published some of his ideas about what machines connected to humans could do in his wonderful work “The Man Computer Symbiosis”. And his vision was that if you can let machines and humans interact, wonderful things will emerge in this symbiotic relationship. They help each other. They do things differently. And wouldn’t it be nice if many people were connected to many machines. In ’62, he was brought to ARPA to form the IPTO, the Information Processing Techniques Office, started funding a number of things. He had been brought out of MIT, at the time. He had spent some time at BBN. And he began to recognize that this is a great domain.

He started talking about the result of men and computers interacting and coined this phrase of the intergalactic network, whether it referred to people’s intergalactic network or machine network was not clear, but let’s give the benefit of the doubt. Let’s assume he was anticipating a network of machines in broad based deployment, nation-wide, worldwide, perhaps galaxy wide that would allow these benefits to come forth.

Shortly thereafter, he brought Ivan Sutherland in. And this is a rather interesting story, as well. Ivan then took over IPTO. And he knew of my networking work, as well. He came to UCLA. And he recognized that we had four-- three identical IBM machines on the campus. They were all similarly configured, running the same operating system. He said why don’t you-- he’s talking to UCLA, not me at the time. Why don’t you put them in a network, in a three node network, and let them interact. And then, I did not get involved in that discussion with the administration. The plan-- the effort was to create a plan to do this, never happened. Never happened because of the administrative jealousies that took place. The idea being you have your machine. You’re using it 100 percent of the time. How could you possibly put it in a network where other people might use it? And I don’t want you to touch my machine anyway. So it never happened, but the idea of a network at ARPA-- the idea was now beginning to germinate in Ivan’s head. And he passed that on to the people that followed him.

Following Ivan, we see Taylor come in, Bob Taylor. And Bob Taylor was enamored with this idea. He was, at the time, funding a number of universities and industrial research centers with a terminal connected to each one of their main machines. And he was frustrated that he needed a different terminal for each machine. He was also frustrated by the new principal investigators who were coming in. Each time he tried to engage new principal investigator that person would say-- the researcher would say buy me a computer. And Bob said, “Sure I’ll buy you a computer.” And then the guy said, “But don’t only buy me a computer, but I want all the goodies that all the other sites have, the graphics at Utah, the simulation at UCLA, the high performance at Illinois, the database at Stanford SRI.” And Bob said, “Look, we can’t afford to give everybody all those resources. We have to share the resources.” Same phrase.

So he decided, we need a network. And so, he brought someone who now had some exposure to networking, Larry Roberts, to come in and make that happen. Now, to get Larry there is another story. Larry was as happy as he could be doing research at Lincoln Lab, running that experiment, for example. And they tried to coerce Larry to come to run the program at ARPA to, in fact, make the network happen. So, just around that period when they were trying to convince Larry, I was back in Boston, and Larry and I were seeing each other. And I remember sitting in his Volkswagen in Lexington in the rain, and he's telling me, "Listen, they want me to come to Washington and leave this wonderful research environment and manage this project. And, should I do it, Len?" So, we talked. I said, "Look, Larry. I remember the experience I had at Lincoln Lab, when they said, 'Try it. If you don't like it, come back.' You have that option. It's a new experience, a new adventure. Go for it."

So, a lot of messages were coming into the same destination node. That means they had to be reassembled at the node before they're delivered, the various packets need to be reassembled back into the message. Well, it turns out that the reassembly buffers were full of partially reassembled messages. They were missing some of their other packets. All the nodes to which that node were connected had messages going to that node, but they were none of the ones that were missing. They were none of the packet that were missing. The packets you needed to complete those that were being reassembled were out here one layer away.

In order to get through, they needed buffer space on his intermediate nodes, but they couldn't release space until they could send it to the destination node. But they couldn't because there was no more re-assembly space left. So, they couldn't release to allow the packets which would release. Crash, down it went. Again, the solution there was to reserve space before you launch it. And we saw a number of other cute things of that sort.

Weber: And talk a little bit, some of the grad students that you have working for you in this period.

Kleinrock: Okay. The grad student story. This is the biggest story. The biggest story has to do with culture and delegation. I'm going to take you way back to my original thesis design in which I said I did not want one node to control everything. It would be saturated; it was vulnerable; it was a bad idea. So, I distributed the control throughout my mathematical and simulation model. Every node participated in a piece of the control. One delegated responsibility out among a group that would then cooperate and collectively do the overall control. Now, let's move fast forward to ARPA.

How did ARPA deal with their principal investigators? Well, those were the golden days again. They would find someone they trusted as being a fine researcher, maybe proven themselves, maybe simply promise, and said, "We want you to do research. Here's money. Go investigate your domain and make something happen." Occasionally, they said, "We'd like it in this particular area, and we'd like this kind of goal. But here's the money. Go ahead and do it." Not a lot of project meetings. Not a lot of site visits. Not a lot of reports. Not a lot of feeding back. We'll delegate to you the job. Here's what we'd like done.

You decide how to do it. Again, the same notion, trusting those who are going to carry out the functionality. Okay. Now, principal investigator needs a job done. Who's going to do it? Well, partly the faculty researchers as well as the graduate students. That's the main source of brain power that's going to make this happen. So, what we did with the ARPANET development was the following.

The infrastructure, the wires, the protocol, the underlying infrastructure of the bits, the lines, the routing, the error control, that was all laid out in the spec, and we made that happen. But the protocol, the higher level protocols, how hosts should talk to each other, how users should interact, how to get telnet protocols, voice protocols needed to be built--certainly the host-to-host protocol--before this network would function. As research principal investigators, we told our graduate students, our software programmers, "Look, here's what we need done. Go do it. We trust you to do it." And they took that, and they brilliantly formed their own organization. They had their own meetings where the principal investigators were not present, decided what their procedures should be, the rules of order, the protocols, the documents they would exchange across many universities. And collectively they developed-- the first thing they did was the host-host protocol.

It turns out that Steve Crocker, one of my software programmers, was in charge of the overall design. He wrote the first request for comments – RFCs-- on the host protocol. They formed a network working group, and they organized themselves. And that overall culture between the ARPA managers, the principal investigators, the graduate students and all those participating, was one of shared admiration, responsibility, respect, ideas going back and forth, and the gratification was not financial. It was: Are you respecting, are you going to use what I've developed? Somebody uses what I do? That's what I want. The challenge problem, making the technology. We're here at the beginning. Let's do it properly, and let's do it in a shared, open, trusted fashion.

And that was the secret sauce that allowed what we now call the Internet to emerge the way it did. And that worked for 20 years. People behaved. There was an etiquette, a protocol, trust. I knew everybody in the ARPANET in those early days. I knew they were good people. We didn't build any security protection. Who's going to do something bad? The community will control that. You know, don't do it. And what happened? The netiquette kind of thing. And so, we did not build in strong user authentication, strong file authentication, which we should have done.

Of course, today in the Internet, those security controls are certainly needed. But had we put them-- had we developed them, the first thing we should have done was turn them off, because the community was the right kind of community to grow. And to make it grow, you need less restriction, encouragement, invitation, rather than saying, "No. You can't do this, you can't do this." Turned it off, and then if we reach a point like we have now, where we do need user and file authentication, crank it up to the extent you need. But in those days, we just wanted to generate new uses, new traffic, new applications, new ideas. The culture-- I can't emphasize how important that whole culture of openness, shared trust, respect, creativity, idea generation, was to this whole development.

Weber: Any particular anecdotes, which would illustrate that situation?

Kleinrock: Well, again, the product of the host-to-host protocol is one example, where this group just developed it on their own. It took them a lot longer than we thought it should. It took almost a year and a half. It should have taken, we thought, a summer. But that was kind of the growth of the process. They were growing the process as well as the product, if you will. And the kinds of relationships it created, these people remained colleagues, friends, co-workers, for their entire professional career. It's hard to think of anecdotes, per se. But it was a very close community, and I can illustrate that with a different kind of anecdote.

In 1973, there was a conference, an international conference on computer communications in Brighton, England, at the University of Sussex. That's when Bob Kahn and Vint Cerf introduced their idea of TCP. And we went there. We were put up in the dormitories. We already had a link to London, an ARPANET link, so we temporarily put a link down to University of Sussex so we would have ability to use the network and connectivity. I came home a day early. I was unpacking, and I noticed I had left my electric razor in the dormitory at University of Sussex. And I wanted it, so I said, "Well, how can I reach these people?" Well, the ARPANET. I said, "What crazy would be on the ARPANET now? Because in London it's 3 a.m." So, I said, "Hmm, maybe Larry Roberts."

So, I had a piece of software on my computer which knew that it lived in a network. Most software didn't know it was in a network. It was Viso [ph?] Sharing Executive. And what you could do, you could type in to that Viso Sharing Executive, _____ exec., say where, and give a name of somebody. I put down "Where Roberts." It took that command, and it logged into every machine on the network. There weren't that many at the time. Looked at the who list, who's logged onto that machine, and look for something with the name Roberts. I was sitting in my bedroom at my [Texas Instruments] Silent 700 remote terminal.

A few minutes later, clickety-clickety, clack-clack-clack, it's printing out. It says "Roberts logged on at BBN at Teletype so and so." So, he was logged in from London to BBN and Cambridge, Mass., and it's telling me this. So, I connected to him in a chat session, probably one of the first chat sessions, and told Larry my problem. He says, "Don't worry." The next day the razor came home. Danny Cohen was there. He lives in L.A. He came home from the meeting and brought it with him. That was probably the first illegal use of the ARPANET. This was a personal use. It wasn't research, wasn't about the protocol, it was a personal use.

Anyway, that's an example of the, you know, the camaraderie we had. It was a magnificent time. I mean, Alan Kay is another example. He was working up with the University of Utah group, graduate student, one of this wonderful group of brilliant, magical graduate students that was creating the software. He came by. He felt very comfortable walking into my office, introducing himself. We had a two-hour discussion. It enthused him in his work and in his ability to reach the principal investigator at another university, and we've had this close relationship. I had breakfast with him this morning, for example. So, but the people happened to be an excellent cadre of people. We were fortunate in that sense, just great, great, creative people.

Weber: That London meeting is where Peter Kirstein had the terminal at his house, right?

Kleinrock: This was not in London.

Weber: Right. Well, in Brighton, they ran the live demo.

Kleinrock: They ran the circuit so you can connect back; that's correct.

Weber: Did you go to his house?

Kleinrock: No. Not at that time. I was to his office a few times, but I've not been to his home.

Weber: Okay. There's a story around that where he put in a password or something.

Kleinrock: No.

Weber: And your lab at the time, or your group, I mean, how much of your time and effort was the ARPANET in that period?

Kleinrock: Okay. So, let me describe the group. This group of about 40 people consisted of some faculty, staff, programmers, most of them graduate students, some hardware engineers, and some researchers. Those researchers were my PhD students working on the ARPANET technology. For example, Gary Fultz, we created this adaptive routing procedure that was so effective, and finally implemented.

And so, I spent a lot of my time with my research, my PhD students. Steve Crocker was in charge of the software group. Mike Wingfield was in charge of the hardware group, which was him and an occasional additional student. So, I delegated a lot of the functionality there. We had meetings, a lot of reports, budgetary issues. But the details of what was going on in the software group, they managed themselves. In the hardware group, they managed.

The research group--of course, that was my domain, and so I made sure that the work was going there, the papers were being published, and the ideas were filtering back into the ideas that these protocol guys were doing. The idea of distributed routing, the idea of error control, parallelization, multitasking, etc.

Weber: But there were other-- I mean, the ARPANET was not the only project in the group… You guys were doing other things, right?

Kleinrock: No. That was the only project. We had, as I told you, ARPA gave lots of money, very good funding. That was my total effort. Other faculty were doing work for NSF, other aspects of-- I don't know if they were the ARPA contracts-- NSF, certainly, AFOSR, other contracts at the time. But no, my--

Weber: Okay. All your people were dedicated to that.

Kleinrock: Absolutely. Absolutely.

Weber: And that was all considered the Network Measurement Center, all that was under the rubric even the software group developing the hosting protocol.

Kleinrock: The host-to-host protocol, right. The name of the project was Network Measurement Center. And some part of it was doing the measurement technology. Another was doing software development, example, some hardware development. And it shifted in time, of course. It started out all developed with the implementation of host and protocol and the network measurement-centered effort, then network measurement center. We only continued until 1975. That's when DCA was given the role of, quote, measurement.

P.S. I don't think they ever made another measurement. And do we have measurements of the Internet today? Do we know what's going on? In those days, we knew exactly what was happening. We knew exactly what its capability was, how far it could go. Now, it's not clear had one entity continued to make those kinds of experimental measurements that would still be effective today. But I know from '75 on, there was nothing else going on in terms of poking the network. And that's when things began to get away, because these latent errors that we're seeing occur even today were not found by pushing the network around and trying to force conditions to extract and expose those problems. Now they get exposed by accident and with a very large user base.

Weber: And once the ARPANET began to be used by real people for real things, true even quite early on, when you're doing these simulations of breaking the network, weren't you directly screwing up services people would want?

Kleinrock: Okay. So, let's look at the history of the use of the network. Until the host-to-host protocol came out, there was almost no use of the network. Most of it was testing, except...

Weber: Remote login.

Kleinrock: Well, yeah. But think about that. You want to remote login to some other machine, a machine with which you have no familiarity, first of all, you have to get an account in that machine. You've got to learn the command language. You've got to learn the operating system and some of the software. It's a nightmare. So, the way it was being used is that someone from Utah, who knew their system, would take a job at another location. Well, not only was this an attraction, this was a necessity, because they didn't know the local system yet. They were much more comfortable using their old friend. So, that generates a minimal use. There was some other work that we were not privy to, where some of the military bases were communicating. They had their own protocol. They didn't need our host-to-host protocol. And data was going back and forth. We can't tell exactly what it was. So, slowly use began, but it was very minimal and by people who really knew what they were doing. So, they expected this in experimental network. If it broke, it broke.

When we finally did the 1972 public demonstration in Washington, DC, a few things happened. Every site brought up their best applications. But that was a kind of local thing to do, you know, bringing up a chess-playing program at MIT. At the demonstration, we were beginning to show the world this thing called the ARPANET. We passed out manuals, said, "Look. Here's a user manual. Go ahead and run a chess-playing program at MIT." And there were some programs that actually used multiple nodes. I remember Danny Cohen and some others put together an air traffic control program. It was a brilliant application. They were going to track an airplane taking off, I think, in Boston, all the way to Washington, passing through New Haven air control, New York air control, on down to Washington. And so, they had simulated an air traffic control system, and as the plane in the simulation passed, say, from Boston, say, into the New York air traffic control, a different machine took over the simulation, as if it was the New York air traffic control. And so, that was a very nice--the program had to migrate the data--it was a very wonderful application. But mostly, they were playing artificial intelligence games. They were playing chess programs, etc.

But I'll tell you what UCLA was doing; it was a really interesting application. We wanted to be able to sit in Washington, in that room, that basement, at a teletype, logon to the machine at UCLA, there bring up a program, ship it across to another machine to be executed and run, and send the results back to a printer right next to the guy at the teletype in Washington. So, the night before, everybody was practicing their demos, Jon Postel happened to be at the terminal, and he logged on to UCLA, bring up a program, compile and execute it and run it at BBN. In fact, it was compiled and executed at UCLA and run at BBN, and results back. Mind-boggling experiment in those days. Today people, you know, an eight-year-old would say, "What are you talking about? It happens all the time." In those days, it was a real test.

So Jon did this, <makes clicking sound> you know, just imagine happening, and he was waiting for the results to get printed at the teletype at the printer. Nothing happened. Did it again. Nothing happened. Then we looked around the room, and another demo that was being tested was MIT's turtle. The turtle is a physical robot that runs around the room with no eyes, and when it bumps into something, it knows it's there, so it avoids it next time. Looked at the turtle, and the turtle was jumping on the floor and bouncing all over the place. The output for the printer had been sent to turtle. <laughs> We had a lot of fun. We fixed that really fast.

But, what happened is, a larger user population suddenly became aware there is this network, and they wanted to try to use it. So, they started using them. So, in '73 and '74, we began to see groups using it. And remember, in '72, e-mail came on. E-mail suddenly became an application that was dominant on the network, because there were not many other applications running. It took over the traffic. So, you're right. There was activity; we had to be, you know, a little careful as to how we took down the network. If we knew why it was going to go down, we wouldn't take it down; we'd just try to get that fixed. But by '75, we weren't doing it anymore. So, that period was a growing period, little bumps out of excitement, and then no more measurement.

Weber: And so, you occasionally crashed something in your experimenting?

Kleinrock: It was an experimental network. It was not a production network. And you know, that brings up a whole other concept. How do networks get created? Well, there's sort of three phases. There are the embryonic networks, which the ARPANET was to begin with, or was just sort of being designed and fiddled with and changed. And then there's a prototype network, which you would deploy to a user group, where they understand there will be failures, but it would be up most of the time. And then when it's proven, it becomes a production network, which the Internet is today. Well, the whole concept is you have to keep feeding that pipeline. Otherwise, that's going to become obsolete, and you keep moving from embryonic to prototype to production. And we were between embryonic and prototype in that period when people expected failures.

Weber: Talk a little bit-- I know there was certainly an interesting relationship between Steve Crocker and Larry Roberts over the spec.

Kleinrock: I'm not privy to the details of that. I know there was something going on there. Basically, Larry, when they were developing this thing, at one point Larry came to look at it, and he said, "Start over. It just is not what you're looking for. It's not grand enough. You're not thinking big enough." And so, they swallowed, they gulped, and they took the larger charge, and they came out with what turned out to be a very effective system. You know, the network control program, NCP. And it survived up until TCP came along. But actually, TCP itself did not come along. It was TCP/IP. It had to be broken into two pieces because TCP would not support real-time traffic, and people like Danny Cohen and a few others recognized, I can't retransmit a voice packet, which is an error; throw it away. If it's data sequence, throw it away. I need this streaming traffic. And if not for that ability to create UDP to run on top of IP, as well as TCP on top of IP, we wouldn't have the kind of streaming traffic capability we have today.

You know, a lot of those events, decisions, accidents in those early days have a long tail in terms of how things grew and evolved. You can look back now and see those critical milestones and turning points. But you can't control them then. The only thing you can do, which we all did very successfully, is keep the generality open, do late binding. Don't preclude things now that you cannot yet anticipate. Keep it open and available so things can be invented and developed.

Weber: And Vint Cerf was also.

Kleinrock: Vint Cerf was in my group. He was working under Steve Crocker.

Weber: So, talk about, you know, what was it like working with Steve and them.

Kleinrock: They were great guys. There was a gang of a few, Steve was the gang leader, Vint was right there below him. People like Charley Kline, Jon Postel. I mean, so, imagine you're the principal investigator working with these guys. They're not an easy group to manage. They're aggressive; they're smart as hell. They were able to get lots of things out of me, and I was happy to do it, but, you know, they had to present a case, and they presented good cases. They needed to travel somewhere, I'd approve the budget, and, you know, it was always a battle. You know, budgets were large but finite, and so we had to decide how to allocate it. But working with them, you know, they were developing the protocol, then the software and some of the testing, in a very, what shall I say, creative way, a very effective way.

And they were running their own shop, this network working crew. And the request for comments effort was all in the domain of the graduate students. They were brilliant guys. I was not on Steve's committee. I was not on Vint's committee. I actually may have been on Vint's committee, I'm not sure, but I was not a supervisor is the point. And they were all doing dissertations about this domain, about protocol, most of them, about correctness, about testing. And they were just another bunch of graduate students, who happened to be doing the software for our ARPA effort. They were wearing two hats. And they managed to merge them as is appropriate. The work you're doing for the project should be your dissertation work, and it worked very nicely. Same for my own graduate students.

Weber: And so, day-to-day, I mean, how often would you meet with most of these folks?

Kleinrock: Well, I met with my graduate students as a group once a week and individually as well once a week, at least an hour each. I met with the software group typically on demand when they needed some things reviewed. It would be two, three, maybe four times a month, not on a regular basis, only when emergencies came up and they needed something in terms of budget, decision point, direction, high-level interaction with the ARPA folks. But I did delegate a lot to them because I trusted that they knew what they were doing, and you know, they were experts in software development, and I trusted them to do it right, and they did.

Weber: And what was sort of the feeling like in the group? I mean, they were very cohesive? A lot of socializing?

Kleinrock: Well, we had a lot of parties, a lot of events, a lot of-- there was some, you know, the software group itself had dissentions internally but not ones that bubbles up to great visibility. But, you know, the software group worked well with the hardware. The theoretical guys worked with them, but the exchange of information was not that strong, and that's always the case with theoretical work. Although we did begin to develop some of the hierarchical routing ideas that were later implemented in the entire Nth router procedure.

Farouk Kamoun, one of my students, and I did that. But they were pretty much separate efforts. It wasn't like this. As a social group, it was like this, but in terms of interaction at the work level, not that strong, because they were doing different kinds of things. Theoretical guys were trying to develop principles, trying to develop performance evaluation, trying to develop methods of looking at things. These other guys were implementing, and developing new protocol concepts at the same time.

Weber: I know you've been involved in a number of different ventures. If you don't mind, talk a little bit about what became Qualcomm, which I know.

Kleinrock: So, the Qualcomm story is an interesting one. I knew Irwin Jacobs when he was a student and a professor back at MIT. Then I came out here to UCLA in 1963, and there were a few assistant professors that started at the same time. One of them was Andy Viterbi. He had worked here on the West Coast. He had gotten his master's degree--or was it bachelor's--I think it was master's as well--at MIT, but he did his PhD here out of USC. And he and I got very close because we were both in a group that was dealing with communication theory and networking and the overlap. So, we had to teach courses in each other's areas. He was more on the analog side. I was on the digital side. Networking communication theory, Wiener, some Shannon networking.

So, Andy and I spent a lot of time teaching each other, sharing each other's courses, because that's the best way to learn. We'd be on the phone every night talking about research, talking about the university. In 1968, Irwin came out on a sabbatical to work at JPL for a while, and he also spent some time down at UC San Diego in that same interval. I forget his exact path. But we were coming home from a meeting in October 1968--no, it was more like about two or three months before that--on the plane together. And we were all very chummy. In fact, Irwin and I published a paper, a very good paper, together, by the way.

And we said, "You know, we ought to form a company. I mean, we all have good consulting practices, well lots of of ideas. Let's get some synergy here." The idea went quiescent.

A couple of months later, I got a request from the Naval Electronics Laboratory, NELC, down in San Diego, to consult for them. But their consulting rates, being government, were very small. And I said, "Look, if I had a company, could you pay us more?" He said, "Yes. We can give you a two thousand dollar sub, for example." In those days, that was a lot of money. So, I called Irwin and Andy and said, "Here we are. We've got a contract. Let's form the company." And so we did. And so we did the usual things about getting started. We had to get some stationery. We had to get a name. We thought about it. And I came up with the name Linkabit-- Link - a - bit. And Irwin and Andy went, "Hmm." And then I realized why they were disturbed. They said, "You put your initials into the name Linkabit-- L-K." I said, "What are you talking about?" <laughs> So, we agreed Linkabit and had decided on the logo, a big L made out of punch paper tape and the code on the tape should say Linkabit. But we never did that part. But the idea of tape and all of that. We decided we needed an office, and I volunteered my home, because my wife, in a psychotherapy practice, had an office there that we could use.

Weber: It's not here.

Kleinrock: It was not here. It was actually about a mile and a half away, on Kenter, 801 North Kenter. So, that office became the first office of Linkabit. We hired a secretary, who, by the way, later left us and married one of the faculty at UCLA. And we still keep in touch with her. My home became a classified facility, because this was a classified contract. Now, to become a classified facility, all you do is buy a proper safe in which to put classified documents. The safe was in the garage. The office was nearby, and we're up and going. And so, we worked on our contract and began to get some others. And then we decided we needed more professional space to bring potential clients. So, we rented a space right almost on the UCLA campus, on Weyburn, which is in Westwood Village in the second story, some nice offices, minimum budget.

We started working there, doing some contracts, wrote a bunch of reports and papers. And then, in April '69, that's just a few months after the October start, things began to get very busy at UCLA. Namely, the IMP was going to arrive in a few months, and I had 40 people I was in charge of. So, I said to Irwin, "This summer, in July and August, I am going to be really busy, because September, early September, that IMP arrives." So, I said, "I won't be able to work on Linkbit during this period." He says, "That's fine." So I went and did that. It came. It got connected, went back, spoke to Andy and Irwin, and they said, "Hmm, Len, you've not been working here. You're out." So, they basically said, you know, "You can't work anymore." I said, "What are you talking about? We agreed." Nope. So, I was really upset. I mean, not so much from Irwin, because Irwin and I were not that close, although our families had been close back east. We actually went camping in Cape Cod with our kids a couple of times, but not like Andy.

For years, Andy was my closest friend. And to have Andy do this was a very disappointing blow. And I had no idea about minority right. It was a one-third, one-third, one-third split originally. I had no idea about minority rights, no legal issue, so I said, "Look, tell me what you think my contribution has been worth, let's sign a document right now, and I'll step out." So, we agreed on a number, a percentage of the company. Sign a document; we all signed it. And that was it. I went off and did the Internet kind of thing. They went off and formed-- took Linkabit, and Andy quit the university, as did Irwin, and went out to San Diego, to La Jolla, to form, basically, Linkabit.

Now, to be honest, if I had the choice somewhere along the line to quit my university position and go down there, I would not have taken that choice. I would have stayed at university, and I'm happy there, to be that way. I was just very disappointed what happened. Well, to continue that story, Linkabit grew very nicely. In 1979, it was bought up by Macom. I received some money as a result of my ownership, which had been diluted by other shareholders. Andy and Irwin were locked into Maycom for a while. The story I hear, and this is now third hand, that Irwin was not moving up to CEO position there for whatever reason. So, when their lockup ended, he left. Irwin and Andy got together, formed Qualcomm, end of story. So, I was very disappointed what happened back then. I'm very happy with the fork in the road I took, and undoubtedly they were happy with the fork in the road they took.

Weber: But that was the first company you've been involved in.

Kleinrock: And a successful company, sure. Yeah. In 1976, I formed a company called Technology Transfer Institute. The reason I did that is I had just published my two-volume work on queueing systems, the second volume of which was largely devoted to the ARPANET technology, the ARPANET experimentation, the results, the data, the protocols, lockups and all the rest. And I felt, in 1976, it was time to get that word out. It was the first document about the ARPANET. First, not document, first book. And I felt it'd be really good to let the industry learn about this quickly.

So, my wife and I decided to form a small company and put me on, on a three-day seminar across the country. And so, we did that. The company was called Technology Transfer Institute [TTI]. We spent about ten thousand dollars promoting this three-day seminar three times across the country. The first one was to take place in Dallas, Texas, at the Fairmont Hotel. And we created these forms. You know, we advertised a little bit. We got some registrants. We had to create the whole business structure to do this. You know, it was nontrivial, how to get notes, how to deal with the clients, invoice them, collect money, get a bank account. So, we couldn't afford a staff to handle the conference in Dallas. I had to do it.

So, I got all the notes and two big suitcases, copies of my own work, and copies of the books, which we shipped. Took them out to Dallas, set up the registration table the morning before the conference. People started coming, and I registered them all, gave them their notes, gave them their badges, pencils, whatever, closed up the registration desk, walked onto the podium, introduced the speaker, <laughs> and started speaking. <laughs> And it was wonderful. It was only 14 people. But the next one a few weeks later was in Washington, DC. We had 60 people. Now, that was a more serious event. Stella came with me. And between the two of us, we did the registration. And we felt it was sort of, what should I say, low class for her to-- Stella Kleinrock to be working here. So, we just said Stella. Okay. So we're doing this. During the registration process, one of the attendees comes out, goes over to Stella and says, "Are you the Stella to whom this book is dedicated?" Because I'd dedicated it to Stella. So, again, our cover was blown quickly. It was very successful.

Came out here to L.A., did another successful one. It was a little smaller than Washington. And we said, "This is a good thing. Now we know how to do it." We made some money. It wasn't a lot. Figured we know other people who are prepared to transfer their technology to the professional world. And I set up criteria that the speaker had to satisfy, two out of the three following criteria. Must have written "The" book in the field, be a great speaker and be well known. I chose Larry Roberts, Norm Abramson, Bob Lucky, Peter Denning-operating systems. So, we put on five conferences in February of '77 and did that twice a year. And then it grew to six, to twelve--it was growing nicely.

And that company eventually took on someone named James Martin, who was the guru in the computer field in terms of exposition, British. And he was being run by a company out of England, but the people who were running it, the wife was afraid to fly, so they came to me with a successful TTI, and they wanted me to take Martin's U.S. conferences. It was all the same conference. It was updated each time, but the same conference--five-day conference, very expensive, eight times in the United States each year. So, we stepped up a very interesting level. It was much larger now. It was much larger domain of technology as well, because it wasn't just the engineering stuff. It was database, it was user interface, it was networking as well, and a whole bunch of things.

So, that grew very nicely. We did very nicely there. We took on, began to do a trade show with John McQuillan, the ATM conference and trade show. We sold that. And the seminar business began to drop off when the economy got difficult. So, in 1998, TTI took over another kind of conference, called Vanguard, which I'd been running since 1991. It was run by a board of advisors five times a year, advanced technology, emerging technology. The idea is members come in and they pay a yearly fee, and they hear about things that are coming down the pike, evaluating technologies they should know about so they shouldn't be surprised. We track technology. So, we continue to do that now. That's what we do only now, very successfully. Run it five times a year. We have many members. We have a board of advisors. This thing was founded by Nicholas Negroponte and Alan Kay. They're still with us. We have a number of others. Bob Lucky's still with us. So, we took that over, and that's what we do now. So, TTI is a very successful startup as well.

So, we left the conversation where Larry's going to consider coming to Lincoln. But meanwhile, Bob Taylor went to Lincoln Laboratory and pointed out to them that most of their funding comes in from the government, and wouldn't it be nice if they allowed Larry to come to ARPA and work on this project. And suddenly Larry-- a little bit of pressure was put on Larry from above. Larry appeared in Washington. And

I don't think he's ever regretted that. I think it was a very good move for him because what he accomplished was just magnificent.

So, Larry appears on the scene, and Bob Taylor basically tells him what his idea is, to get a network to connect together his principal investigators and their machines. And then Larry began the process of thinking about it. He spoke to me about the technology issue and wondering how he should get going. He put together a task force to write what the specs should be, spoke to Wesley Clark sort of in an accidental conversation, and Wesley came up with a critical idea for the Internet. Mainly, don't burden the host processors with all the communication load of this data network manipulation, packetizing, error control, routing control, etc. Rather, offload it to a separate machine, a communications processor, if you will, which would talk to the host machine.

But these communication processors, which later came to be known as interface message processors, IMPs, the IMPs would talk to each other and provide the backbone network for this communication. It's a brilliant idea. So that was put into the spec. And some of the other issues about response time were settled. We had a meeting of a few of the people involved with this, and each one had their own view. Herb Baskin was there, and he basically said he wanted good timesharing. He wanted this to look like a timesharing system to him. You had a terminal. You want to get the remote machine. It should be as if you're accessing your local timesharing system as far as for login and performance and all of that. So, he said, "The response time can't be more than half a second." So, he banged his fist on the table, and the spec said, "Response time for short messages no more than half a second."

So, we proceeded to do this. I was there, and I pushed the fact that this is an experimental network. We need to be able to measure what's going on. We need software to do that. We need to be able to load the network the way we want, not the way the users happen to load it. So, we need artificial traffic generators and mechanisms and be able to trace packets that pass through the network, etc. So, we said there should be network measurement software built into every switch. And reliability-- interesting question. The way the telephone company specifies reliability is to maybe four or five nines. It's good 99.9999 percent of the time. And that's an interesting fiction, if you will, because those numbers in the telephone network only apply when the network is up. They don't count the downtime. It should only break that kind of fraction of the time, when it's down. Anyway, it's a funny measure. So, rather, we took a much more pragmatic approach. We said, "Look. It should be if anything breaks in the network, it shouldn't stop the other things from communicating." So, that defined what we called a two-connected network, which means between any two nodes, there are two independent paths. So, if something on one path breaks, you always have the other path. We put a number of these specifications.

Weber: And this was your meeting in Washington for this?

Kleinrock: Yes. This was a Washington meeting. I think it was Washington. I'm trying to recall now, because we had more than one meeting. I think the critical meeting took place in Washington, and it was 1967, I believe. Larry took that, and he formulated--

Weber: And maybe you're about to tell me: What document came out of it?

Kleinrock: That was just input to Larry at the time. Larry then put together an alternate plan. He got the plan approved by the director of ARPA. From that, he created the specification, which was to go out for a bid, a request for proposal. Created that thing. It went out, I think, in July of '68 or so, went out to industry, to please create a network, create an IMP, however you like, typically take an existing minicomputer, modify the software and hardware, according to the guidelines that this request specified and add your own innovation, deploy the IMPs, bring up a four-node network to begin with, lease the lines from AT&T, putting these switches at the ends instead of the circuit switches from AT&T, just use their lines and put our switches in, and make it happen. Went out to industry. There was a bidding process. A number of people bid, notably ATT and IBM did not. More than one group bid the same minicomputer as the basic machine, a Honeywell DDP-516. And the reason that machine was so attractive is, it was a state-of-the-art machine with 16 levels of interrupt, which meant that one process could interrupt another process. Well, think of that as incoming packets or messages or interrupts. This machine has to be very reactive and very interruptible and multitasking. That was one of the main features of this machine.

Oh, by the way, Wes Clark also said this processor, this communications processor, now called an IMP, should be able to be standalone, unattended. No human interaction with it, put it in the closet, and it should run, which means no electromechanical parts which can fail. The IMP had no rotating machinery in it at all, and only the first few had paper tape readers to load the program in. And once they were connected by the backbone network, they would send that new releases through the network and no longer through the paper tape. The spec went out.

Finally it was awarded to BBN around Christmas time of '68, and it was there-- the timing was such that we wanted the first node to appear in September of '69, less than nine months later. And it was decided that UCLA would be the first node--because me and my group had the expertise to do the network measurement center--stress the net, work test it, measure it, and to think about what the fundamentals were. The second node was to SRI, third node University of California Santa Barbara, and the fourth node in December of '69 would be University of Utah. BBN got this job. They put together a wonderful team, a magnificent team of people. They worked on it, and they kept things pretty quiet.

Meanwhile, at UCLA, we know we're going to be the first node. And the piece that BBN is not responsible for is the communications link between the IMP and our host time sharing computer. It was up to us to make that happen. But BBN had not yet released the spec on how we could talk to that IMP. We kept asking them for it, and they were very slow in giving it to us. So, we were falling behind. Finally, we got the spec, we started implementing the hardware and software version of that. We were told BBN was falling behind, they would not be able to send it out until Labor Day weekend. We breathed a sigh of relief, kept working like mad, and the son of a guns, BBN, air freighted the darn IMP out to us, and it arrived on August 30th, on a Friday, just before the Labor Day weekend. We worked on it over the weekend.

Comes Tuesday, the day after Labor Day, the big event. Switch, host, will they be able to talk to each other? And so, it became an event with some notoriety. Everybody came to see this happen. We had people from ARPA, people from the computer science department at UCLA, my department, people from the engineering school, people from Honeywell, people from ATT long lines-- we were going to use their lines. People from GTE were going to use their local lines. The administration of various organizations. And everybody was ready to point the finger at somebody else if it didn't work. Happily, as soon as we connected them, the bits began to flow back and forth, just bits to show that the link was working. So, now that was fine. And so, we continued to work on it. We didn't have any important records of that event, but there were a lot of people there to witness it. So, the plan continued in October.

The second IMP was delivered to SRI. A little while later in the month, they connected their host computer to it. Then toward the end of the month, we decided, well, let's communicate between the first host and the second host. Let's see if we can get the sense that we can log on to their timesharing system and make us look like a terminal to them, and make their machine look like a nearby timesharing system to us over a remote line. So, Charley Kline, one of my software programmers, and I were in the room, 3420. Bill Duvall was up at SRI, Stanford Research Institute, and we wanted to get this login to take place. At the login, all you have to do is type L-O-G. And the remote machine is smart enough to type out on your terminal the I-N to complete the word login.

Now, to make sure this is going to work, we got a telephone connection between us and the SRI folks. So, Charley typed in the L, and he said, "Did you get the L?" And Bill said, "Got the L." Typed in, "You get the O?" "Got the O." Type in the G, "Get the G?" <claps hands> Crash! The SRI host crashed. And noticed, it wasn't our host. It wasn't our IMP or the other IMP or the long haul line. It was the SRI host. So, it had nothing to do with the network. It was a memory overflow on the SRI host. So, we fixed that, a little bit of software, and got it to work properly shortly thereafter. This happened. The log entry that Charley made was at 10:30 at night on October 29, 1969. That's the only record of any sort of that event, the event being the first message ever on the Internet. No camera. No voice recording. No serious documentation, except an entry in the log. And the log simply says, "Talked to SRI host to host." Very important event. Next month, Santa Barbara came up, three-node network. Following month Utah. We now had a little toy network to play with and test some other functionality, like the routing procedure, like the error control, and a few other issues. So, we tested it. We were able to break the network at will. Once we broke it, we told BBN what we saw. It was their job to fix it. And they began to deploy more networks. After a three-month delay, they started loading up.

By December, we had ten nodes up. We had a cross-country link, all the way across the country from UCLA to the East Coast, Cambridge, Mass, area. We began to continue to stress this network. As a network measurement center, that was our idea. We'd stress it, easily break it, tell BBN, and BBN would say, "It's going to take us six months to fix those things," typically. And we said, "Well, a simple thing. Fix it. Figure it out and fix it." I say figure it out because they did not give us the source code for the IMP, so we couldn't tell exactly what was happening. We could see externally what was going on. A big litigation took place shortly thereafter, where ARPA said, "You can't keep that code secret from the investigators."

The government won. The code was released, and we continued to measure the network. And any time we broke the network, not only would we tell them it broke; we'd tell them how to fix it. Still took BBN six months. And so, you can say, "Well, what kind of priorities did they have?" And the interesting thing is, Frank Heart, who was in charge of the project at BBN, it was his job to keep this network running, a lot more than fixing the bugs we happened to find. So, in some sense, his priority was different from ours. There was contention there, certainly. You know, we would say, "Fix it. Here's how to fix it. Do it." And it had to be scheduled; it had to be done professionally. And so, we were irritated. Frank was trying to maintain an orderly way to handle all this. And eventually, they began to fix these bugs more quickly.

Weber: You knew Frank from back at MIT, right?

Kleinrock: Yes. I'd bumped into him at Lincoln Lab, but he and I never spent much time together, because he was in the systems area, and I was in the computer area. But, of course, at this time, we got to know each other very well. And we interacted quite a bit during the preparation days for the first delivery and through the whole rollout of the deployment of the ARPANET.

Weber: And what was it like working with him?

Kleinrock: He was hardnosed. He was, what shall I say? He protected his domain and wanted things done in an orderly way, which I admire, but not as, if you will, immediate, aggressive, effective way as we felt could be done. He was difficult to deal with. I understand the constraints he had. He had a team. He had a contract. He had money issues, and he had maintenance issues. He was deploying nodes. We were doing basically high-level things of trying to make a network work properly for the long run. So, he was not the easiest guy to work with. But my hat's off to him. He delivered in eight months something you could never do today. It was a new product and new technology, a new set of users. He delivered it online, on budget, working. You couldn't do that today, just couldn't. So, he cut through a lot of red tape somehow at his end, which we don't see, but I'm sure he had his own definitely to deal with in his own organization. And, you know, he deserves a lot of credit for making that happen. He had a great team, but he was in charge of it. And the team leader gets a lot of credit for making it all happen.

Weber: So, what sorts of measurement things did you have? Talk about that, what you were measuring and what was built into the IMPs at that time.

Kleinrock: What was built into the IMPs were the following kinds of things: The ability to generate artificial traffic on demand by the network measurement center. We could send messages or instructions to various IMPs to turn on and turn off and create different kinds of traffic destined to various nodes. The IMPs throughout the network would collect measurements in a distributive fashion, gather them up into measurement messages and send them back to us. And then we would analyze what happened and observe what happened. We could create messages, which had to go from one node to a particular other node, and trace the history of those packets as they went along their path. We could just measure what was going on in terms of who's up, who's down. It was not our job to measure the ongoing traffic or the response time of the users. That was the kind of maintenance thing; BBN was responsible for that.

Weber: Network operation center. So, the provision is they were doing the hands-on system administration.

Kleinrock: Network operations.

Weber: You were measuring the capacity.

Kleinrock: The capacity, throughput, response time. If something broke, like if a link went down, it was their job to get it back up. You know, that kind of maintenance thing was not our interest. And so, we did a number of things. But there's an ironic side to this. There was a time when we discovered re-assembly lockup. That was the name of the lockup, the network locked up. Why? We started making measurements. Network crashed. We told BBN, eventually brought it back up. Made more measurements, network crashed. So, BBN said, "What the hell are you guys doing?" And we said, "What did you build into this thing?" Turns out we found out what was going on. There is a-- no, this was not re-assembly lockup. This was another one. This was a hardware thing. I'm sorry. I misnamed. There was so much storage in the IMPs for collecting packets as they came in, putting them back into messages and delivering them to the host. And that was the re-assembly space. Now, you have to ask yourself, when a packet comes in, it's got to find the right place in that memory to go to so it joins its other packets in that message. You need pointers. Well, how many pointers do you need? And the answer is: What's the maximum number of messages you may have to re-assemble at the same time? So, you take the size of the memory, and you divide it into how many messages could you have. Well, there are big messages and long messages. One packet message, two packets, three, up to eight packet message. Well, one packet messages don't have to get re-assembled. They get delivered immediately. So, the smallest message is two packets. So, you take the buffers that you have, the storage, divide it into as many two packet groups as you can. That's how many pointers you need. You may need less, but that's the max. Fine. Everything was working.

Now, one day, we noticed, when making measurements, down goes the network. Turns out, BBN had upgraded the IMPs and doubled the size of the re-assembly, but they forgot to double the number of pointers. So, packets would come in, they'd grab the first free space if it was the first packet, and reach the point where there were not enough pointers to tell us where that packet was. So, the message got corrupted with this dead stuff, which couldn't go anywhere, and eventually, all the re-assembly stuff disappeared. Network went down. That was not a logic error. That was a kind of implementation error, and a very natural one when you think about it. Who says, "Okay. Let's remember in the future, if you do something good like double the memory, we have to increase the number of pointers." That was kind of ironic. Our measurements caused that but exposed a problem that needed to be exposed.

Weber: But they were on there. There must have been times when there was some interference between the two efforts.

Kleinrock: You mean BBN and us?

Weber: Yeah.

Kleinrock: Oh, yeah. They didn't like us finding fault, clearly. We had to create odd situations to bring the network down. We would send a lot of traffic to one node. Well, that's called a hot spot scenario. That would often stress some aspect of the software or the hardware, as you saw a moment ago, that would take it down. But that's exactly what you should be doing when you're trying to experiment and find out where the problems are in the network. We have store-and-forward lockup. That was one that was predictable. As a packet comes into a node, it's got to be store-and-forwarded to another node. Well, before you send it, there's got to be space to receive it here.

So, what happened is there was traffic going this way. This node got full of store-and-forward traffic going that way and the other way. So, in order to free up store-and-forward, that's trying to go ahead, you had to send the packet here. But there's no space here until you release one of them. To release the packet, it had to make it this way. But you were both boxed up. You couldn't receive anything from your neighbor and thereby release something for the next neighbor's packet. So, they got boxed up and they couldn't move. Predictable. You can fix that. There was reassembly lockup, as I mentioned a moment ago. Reassembly lockup was another cute one.

Weber: When you say you can fix the store-and-forward block, how?

Kleinrock: You make sure that you don't launch things until you know there's enough store and forward. And that's the way we solved this re-assembly lockup as well. Re-assembly lockup had the same problem. A lot of messages were converging on the same node. So, in addition to these startups that I was involved with, there was another one with a PhD student of mine. We created a company called

Nomadix, which was one of the first and most creative nomadic computing startups, which has now basically been bought up and run by Docomo out of Japan. It's a worldwide deployment of Wi-Fi access in a very simple way for the user and very flexible and dynamic. It automatically configures your machine, accepts you as a friendly in a foreign environment instead of as a stranger. So, it does all of that for you, successful company there as well. But the thing I did want to do is talk about what happened after the initial ARPANET got deployed and my involvement in that.

As part of the ARPA funding, under Larry and then under Bob, there was a large effort on satellite packet radio and then ground radio packet switching. And there I deployed an army, and a continual army of students, working on the essential problems of packet radio and things that led to the Wi-Fi access, led to today's wireless cellular networks. Some of those are really wicked problems. And even today, mesh networks are not understood and multi-hop mesh networks, and don't deliver the kind of throughput that people think they might. And there are some fundamental reasons for that. But early on, when Norm Abramson first introduced the idea of Aloha, he analyzed it simply. A student and I, Simon Lam and I, analyzed it deeper, were able to show for the first time that Aloha by itself is fundamentally unstable. And one must introduce measures to stabilize them. We showed him how to do that.

Then, when Norm Abramson again introduced CSMA, he introduced that at a packet radio meeting, an ARPA-run packet radio meeting, and he suggested what the equation between input and throughput would be. I was sitting there with one of my then PhD students, Fouad Tobagi, and we looked at that, and we said, "No. That's not right." On the airplane home, we got the exact expression and found the throughput for CSMA, and then elaborated quite a bit on the various related protocols for CSMA, which essentially led to the efforts that we now call Ethernet. That was the beginnings of the understanding of how the protocol should work, stabilizing it, predicting it, introducing hidden terminal issues, etc.

So, some of those that work with your large number of students, who now populate various universities and research centers and startups around the world, sort of launched this whole wireless industry in the data world. That's been a very, very rewarding experience. Working with graduate students I taught is the wellspring of my life, and it just produces good results, and just keeps the field vibrant and solving problems along the way. Those challenges that you have to keep reaching. And, you know, the packet radio effort was big as an ARPA funded effort in the '70s. The trouble is, at that time, the technology led to too high a cost, too heavy a device to carry around, and too power hungry a device. Come the 1990s, you get to revisit it again. Now the technology eliminates a lot of those problems--the size, the weight, the power, the cost--and leads to a lot of the mobile revolution that we're seeing today. The interesting thing is some of the same problems still exist, are being reinvented and rediscovered, and still come to plague us. So, that's been a very interesting ride. The kinds of research I've been doing since then expand beyond that some into intelligent agents that you deploy in a network, and they perform a lot of the work for you in an optimized fashion. Some peer-to-peer networking. I've got some beautiful results in that field. I won't bore you with them now, but the results that have the quality that Shannon basically introduced me to with that. When you get a result, try to interpret what it means. And, in fact, the particular result I have in mind involves entropy as well as fairness as well as some beautifully simple results from some very complex problem statements.

Weber: You can't tease the audience like that. What are you...

Kleinrock: So, let me tell you what the result is, and I hope you understand it. Consider a peer-to-peer network that wants to do file sharing. So, each node has some files, and it wants to share them with others, and it wants to borrow from others. So, the users come in, they look for a file, and they pull it back. Now, suppose you have a limited storage for files, so you can only store so many. And let's suppose there's a certain rate at which this population of users wants access to those files. Now, you want files, as a node, but you don't want to share yours, because that's an effort on your part. You have to store them, and you have to deliver them. So, the question is: Which files would you like to store for yourself? You might say, "I'll store the most unpopular files. Then nobody's going to bother me." So, that's a background to the kind of research we did. We said, "Well, look. What is the optimal way to store files so that the average time to find a file in this network is minimized?" And we use in this assumption a flooding technique to find the file. So, you ask a neighbor. If he doesn't have it, he asks all of his neighbors until it's found. Then it comes back. Finite number of files, a rate at which each file needs to be stored. So, the question is-- you can store more than one copy of a file across the network. What should be the ratio between the rate at which a file needs to be accessed and how many copies there are, given there's a finite number you can store? Well, we solved that problem with some assumptions I've mentioned. And the answer comes out very simple. The number of copies you store should be proportional to the rate at which you want it to be accessed, which by the way is quite reasonable. It's intuitive with reason. By the way, the same concept does not work in the way you assign capacity in a data network. I solved that in my dissertation back in 1961, and it turned out to be related to the square root of the traffic rates. At any rate...

Weber: When you say proportional to the files, in what proportion?

Kleinrock: Well, suppose file one is accessed twice as often as file two. Then there should be twice as many copies of file one as file two.

Weber: Literally.

Kleinrock: Literally.

Weber: For one kind of relationship.

Kleinrock: Now, if you do that, you find that no one can game the system. If you say, "Well, I'm going to store those unpopular files," what happens is there's very few copies of those unpopular files, so you're serving a very large community of people who want it, and it exactly balances out. So, every node has the same load. Everybody's baud is the same. Nobody wants to play games. And when you're all done, it turns out that the time to find the file is related to the entropy of the arrival rates, demand rates. Shannon's old formula. It happens just to be a quirk. We understand because it has to do with the way you search trees, and Huffman and Shannon talked about that in terms of the entropy. So, I think that's enough of a glimpse of this beautiful result, which had some wonderfully and intuitive and happy properties if you pose the problem correctly. I can't tease the audience, huh? <laughs>

Weber: And this is recent research?

Kleinrock: This is in the last five years, yeah. I'm working on a number of things right now. I'm working on some network routing technology right now. There's a thing called any-path routing in the wireless networks. If you send a packet, many people hear it. How many of them should forward it? And how many pairs should be allowed? And we address that issue because you've got some free reception, and you ought to take advantage of it. And so, we decided exactly how you can organize it so this any-path instead of single-path routing should work. It's getting-- we could put some lights on in here, but this lighting here's not great. Okay. Should I talk about the reluctance of people to use the ARPA network?

Weber: Sure.

Kleinrock: Let me see how we're doing. Oh, we're doing okay. So, what's really interesting is, as soon as we started to deploy this network, the ARPANET, in 1969, we also wanted to make sure that the principal investigators participated by putting their hosts on the network. Well, that took some convincing. We went around--Larry and I did this, in fact--we'd call them, meet them, and go to a principal investigator and say, "Look, there's this thing called the ARPANET that's being deployed. Why don't you connect your network--excuse me--why don't you connect your computer to the system?" And they'd say, "What are you talking about? My computer? Me and my staff and my students are using it a hundred percent of the time. Why should we put it out there for somebody else to log in remotely and steal some cycles?" And we'd say, "But you get to use their machines as well, to get access to other resources." And they almost uniformly said no. Well, I kept twisting their arm, and I said, "Look. How many teletypes would you possibly allow on?" I tried to extract a number, and they would come, well, maybe two teletypes [ph?] worth. And how much would you use the network? Maybe four teletypes worth. I got all these numbers, I created a traffic matrix, and I published it in the paper until they were sort of a little bit committed to living up to those numbers.

But what really convinced them was very simple. Larry went around and said, "You're being funded by ARPA. You will join the network." And they did. And as soon as they were on a network, they saw the benefits. It was so easy to see once you started to play with it. I mean, there's a whole other world of other researchers and their results and their services, their applications, their hardware. And you want to exchange it. And you're certainly happy to give up some of your cycles. And as we know, that's a huge inducement on the Internet today. Same story as the UCLA three-node IBM connection. People just didn't want to do it. It failed there; it worked on the Internet. In both cases, pressure did it.

In the ARPANET, pressure did it. UCLA, there was no pressure. So, what's interesting is, as that September date was rolling around for the first node to be brought to UCLA, UCLA was all excited about this. So, one of our PR folks came to me and said, "Let's put out a press release about this event that's going to occur." So, he interviewed me. He put out a press release. He quoted me, in which it says, "There's this nationwide network coming. UCLA will be the first node." And it described some of the uses as to how you can take a program from here and process it here and use a resource from here and get data from there, and, you know, take advantage of the expert parts of each unique resource and bring them to bear to solve a problem.

And later in that press release, which came out two months after the September 2nd date--namely, it was July 3, 1969--there is a final paragraph in this two-page press release in which I'm quoted as saying things like, "This network will basically provide access to computer utilities, will be as easy to use as your telephone or electricity and as invisible as that. And you'll be able to gain access from your homes as well as your offices anywhere in the country."

So, and there's a few more quotes there. The interesting thing is that I did predict a lot of what has happened. I basically said it will be invisible, it'll be available always, all the time, from everywhere. It'll be, any device should be able to get on easily. And you ask yourself, have we achieved that part of the vision? And the answer is no. The Internet is not invisible as electricity is. Those two prongs in the wall give you electricity, you don't know how it works, you don't care. It's always there. Internet, you have to configure things, you have to go through a protocol, you have to log in an account.

It's in some ways much too much in your face. And we're looking forward to a world and an infrastructure and a capability which will make it as natural as I'm talking to you right now, where speech will be the interface, with gestures, where displays will pop up. We'll have deployed technology in our environment. This room will know I'm here. It'll know what my profile, preferences, privileges are and give them to me. Holograms will pop up. So, that invisibility is not here yet. But, what I missed in that vision back two months before the ARPANET began, was that my 99-year-old mother, before she died, was on the Internet. I missed the social side. My vision going into the research and for a while was it's about computers talking to each other or about people talking to computers, not people talking to people. The first time I realized that was when e-mail came on and took over the traffic of the Internet. I said, "Ah! It's about people exchanging ideas, forming communities, growing, shrinking these communities, chatting, etc." But I missed that in those early-- but I had a piece of the vision going. You can think of the computer utilities part as the web-based IP services we have today. You can think of the anywhere access, any time access and the ease of access, and the global, the ubiquitous, the global access. So, parts of it were there.

Weber: Were you--I'm sure you were--familiar with some of, like, Engelbart's Vision for augmentation and that sort of thing?

Kleinrock: Oh, yes.

Weber: What was your reaction to those ideas at the time?

Kleinrock: Well, at the time, Doug's ideas were clearly way out there. And the problem with Doug is that he had a very hard time articulating those ideas in an understandable way. He was sort of always still thinking about the possibilities while he was trying to tell you what his conclusions were. So, he didn't get the recognition at the time for his augmented reality ideas. As of late, he's certainly getting the credit. But he came up with some wonderful concepts that were not appreciated at the time that he was expositing them. Certainly the mouse, that, sure, that was clear winner and recognition immediately. But those other ideas of his have taken a long time to deploy, be appreciated, be implemented and understand the power. I mean, he had the whole concept of hypertext long before the web appeared as did other people.

Weber: <Inaudible>.

Kleinrock: But way back. Go back to Memex [ph?]. I mean, the idea of a repository that could be accessed anywhere, easily, it could be automatically indexed. But Doug was recognized as one of the leaders all along, but the implementation of those ideas was delayed because the technology wasn't there, and the articulation wasn't crisp enough.

Weber: So, when he talked, people basically had trouble understanding what he was describing?

Kleinrock: Well, I think, I won't say the masses--certainly the masses had trouble--but even a large number of the professionals at that time had trouble. There were few of the people who understood the world in which he was thinking and living that worked very well.

END OF INTERVIEW

