Oral History of Robert Metcalfe

Interviewed by:
Len Shustek

Recorded: Nov. 29, 2006; Jan. 31, 2007
Boston, Massachusetts

CHM Reference number: X3819.2007

<u>Session 1: November 29, 2006</u>

Len Shustek: I'm Len Shustek, and today is November 29, 2006. We're here in Boston with Bob Metcalfe, who has graciously agreed to sit for an oral history for the Computer History Museum. Thank you, Bob.

Robert Metcalfe: Thank you.

Shustek: Let's start as early as we can. Why don't you talk about where and when you were born, what your family situation was like, what your parents did, and so forth.

Metcalfe: I am a Viking American. My grandparents arrived in New York City around 1900 from the four Viking capitals of the North Atlantic: Oslo, Bergin, Leads and Dublin. They intermarried, and then they moved to Brooklyn and had children, and then my parents were born: Robert Metcalfe, my father, and Ruth Metcalfe, my mother. I need to mention that my grandmother fought organized crime on the docks of New York. She worked for the waterfront commission of New York and so we were constantly regaled with mafia stories as we were growing up. My father was an engineer, but a technician, actually. Eventually, after 30 years, they gave him the title "engineer". But he was a union member and he refused to leave the union, so for 30 years he was a technician and finally became the only engineer in the union. Then he achieved his two life's goals. He and my mom had two goals. One was to retire and the other was to send me to college, the first member of my family to attend college.

Shustek: What sort of engineering or technician work did your father do?

Metcalfe: My father was into gyroscopes, so he was a test technician for various gyroscopic platforms for the rockets, for helicopters, submarines, destroyers. He once worked on a gyroscope that kept the plane level while the machinegun was shooting, so that the plane didn't roll over in the recoil. In fact, it was his engineering, or his technical work, that got me into engineering.

Shustek: Did he involve you in projects when you were a kid?

Metcalfe: My father had a set routine. He got home every day at 5:30 sharp. We had dinner at 5:30 every day. A different one, although it was the same one each day of the week. We had spaghetti on Friday, I remember. My father was Catholic, so he would have spaghetti and fish sticks, and my mom and us, we were Lutheran, so we had spaghetti and meatballs. He had a shop in the basement where he had lots of doodads; tools, wrenches and various machine parts from work, and an old oscilloscope. He even briefly started a business called BAM Electronics, Bailey Abramson Metcalfe, repairing what came to be known as televisions by replacing the tubes. I remember the AU6 tube. That was one of the bigger tubes. So in this basement, I would hang out, and for a time had a number of broken televisions that I worked on, until one day my father came home from work and found me unconscious on the floor of the basement. Because I had reached into the back of the TV and inadvertently found the high voltage -- the wire that goes into the tube. I had found it, and he found me unconscious. He then removed the televisions on that day. But the story goes-- and this is one of those stories that you're not sure it actually happened, but I've been telling it for 30 or 40 years, so it must be true—that in fourth grade I was asked to write one of the regular book reports. Being the way I am, I left it to the night before, and I had not read a book. So I wandered down to the basement in my dad's shop where we built things-- I'll get to what we built in a moment-- and found a book on the shelf. It was an electrical engineering textbook written by professors at MIT. I decided to make this book the subject of my book report due the next morning, so I skimmed it quickly, noting that it had been written by these professors at MIT. I remember writing the book report. I remember vividly writing "This book had its high points and it had its low points, but on average, it was an average book." But then, just to be sure that my teacher would be motivated to like it, I believe I ended this book report by saying, "And someday, I intend to go to MIT and get a degree in electrical engineering." Which I did, so it was an early prophecy. We believe this occurred in fourth grade, but I have no way to prove it.

Shustek: What were the sorts of projects that you built in the basement?

Metcalfe: The way I got into computers was that my dad and I built a train system. We took one of those 4x8 pieces of plywood and we painted it green and we slapped down some big Lionel tracks on it. Put various controls with these toggle switches that my dad brought home from work, and these little neon bulbs that my dad brought home from work. We put them on Masonite. We drilled holes, put the lights in it, put the switches on it and built this tracks and mountains, the things that people do. Then in 8th grade I was asked by a fantastic science teacher, Mr. Gosher[ph?], who really got me into science, to do a science project. I decided to build a computer.

Shustek: What did you know about computers at that point?

Metcalfe: Nothing, but I thought a good start would be, I started designing an adding machine. But I didn't learn binary arithmetic until the month after this project was turned in. I began building the adding machine with the toggle switches. There would be a row of toggle switches labeled one through N, and another row or toggle switches labeled one through N, and then a row of neon lights leftover from the train set that were labeled two through 2N. I began figuring out every combination. So instead of binary, it was unary arithmetic. And it was done with relays, so on the inside were these relays that clicked. Each one was this big, I remember. I started figuring out every case. In the end, I could not build an adding machine bigger than three switches; one, two and three [in one row], one, two and three [in the other row], and then two through six. If you turned two switches up, the sum would appear by lighting one and only one of the neon lights. I couldn't get to four, because there were too many cases for me to keep track of. The piece of paper was scotch taped together and it had every case. I wired it, soldered it together. When you clicked the switches, you could hear the relays all settling down and displaying the results. So my science teacher, he's the one who called it a computer instead of an adding machine. I remember the grade I got.

Shustek: Which was?

Metcalfe: A+++H superior. Apparently, he began grading the papers, and by the time he got to mine, he had given out all the other superior, so I got A+++H superior on building this computer in 8th grade, which was, by the way, 1959, five years before the IBM 360.

Shustek: Did you compete in the science fairs run by Westinghouse at the time?

Metcalfe: No. This never went big time.

Shustek: What about your mother? What did she do?

Metcalfe: My mom was a housewife for most of my youth, but then she became the secretary at my high school later on. It may have been after I left. Yeah, I think I had left the high school by then.

Shustek: Any brothers or sisters?

Metcalfe: I have a sister. She's 2 ½ years younger and she is a nurse, has been her whole career, a psychiatric nurse.

Shustek: Do you think you knew at the time of that science fair project that you were destined to be a scientist or an engineer?

Metcalfe: Yes, I was going to go to MIT and get a degree in electrical engineering, because I had written that. I owed it to my fourth grade teacher to make good on that. My best friend, starting at about '59 was a guy named Ron Rosenbaum. We lived near each other and we became fast friends and arch competitors in school. His mom was an English teacher in the high school, so he decided to be the literary, literature articulate one, and I decided to be the math and science one. At the graduation in June of '64, he got the award for French and English and all that stuff, and I got all the awards for physics, chemistry, math and so on. He went to Yale and I went to MIT.

Shustek: What schools did you go to before college? Were they public schools?

Metcalfe: Yes, K-12 at the Bayshore School System in Bayshore, Long Island, New York, and so I'm a graduate of the Bay Shore High School and a member of the Bayshore High School Hall of Fame. Ron Rosenbaum and I both made it.

Shustek: When the time came to apply to college, did you apply to more than MIT?

Metcalfe: In my junior year in high school, I applied to MIT. My recollection is I was accepted, but was advised not to go, so I stayed my senior year at Bayshore High School and then went a year later.

Shustek: Do you think that was a wise decision?

Metcalfe: Yes.

Shustek: Why?

Metcalfe: Because my senior year in high school was fantastic. I got a job at Sears Roebuck in package pickup. I got a motor scooter in order to go to the job. I took typing. I took mechanical drawing. I took art. And then I took calculus and all those other things you're supposed to take. It was a good year, worth doing.

Shustek: Were there advanced placement courses in those days that you were taking in preparation for MIT?

Metcalfe: No, there weren't. I took calculus in 12th grade, and that was about as advanced place as you got. But then I took calculus all over again. I decided not to advance place. If you got an 800 in your math, which I did, at MIT you could either move on to the next calculus course, 1802, or you could take a special version of freshman calculus, and I opted to do that. It was called 18... Eighteen is the course number at MIT meaning mathematics and 01, 1801. This was S. To get into 1801S, you needed to have 800 on your math SAT's, I guess it was. I thought I would do that. I walked into a room full of hundreds of people! It was very discouraging. But I did okay. But I wanted to do calculus again, because once again, someone wisely advised me that I wanted to learn calculus the way MIT taught it in preparation for the years that followed.

Shustek: Had you applied to any other schools, or just to MIT?

Metcalfe: I don't know for sure, but I feel that I was rejected at Harvard. The reason I feel that is that I was later accepted at Harvard several times, and each time I felt a tinge of revenge coming. But I'm not really sure I actually applied. I probably did and was rejected.

Shustek: What was it like going to MIT with a room full of other students that also got 800 on their boards?

Metcalfe: In those days, there would be a freshman quiz every Friday and it would rotate; calculus, chemistry, physics, biology. Every Friday, the freshman quiz. I remember the first freshman quiz was math, and I was in this 1801S program. I got the test results and I had gotten 25 out of 125, so I began planning my return home. But then they introduced the notion of class average, and I was then told that the class average was 17. So my 25 didn't look so bad after all. Then I looked over and sitting next to me was a fellow freshman, Rich Schroeppel. I'll always remember Rich. On his paper it said 125, so I knew I was swimming with some sharks there. I was doing all right, although I was not the superstar that I had hoped.

Shustek: MIT has a reputation for undergraduates as being a pressure cooker. Did you feel that kind of pressure, and how did you bear up?

Metcalfe: I loved MIT from the very beginning. Enjoyed every minute of it. I have a few regrets, but not many. The pressure, I loved. There was competition, and we worked very hard, did problem sets into the night, lots of all-nighters. I just loved it. The mistake I made is: in my sophomore year; I got a job, a full-time job, because I wanted to pay for my own way. The scholarship MIT gave me was de minimus. I got a job, initially at Raytheon in Wayland as a computer programmer. So here I am, full-time student, varsity sports, full-time job, social chairman of my fraternity. I don't know when I slept, but I am proof that you can learn while asleep in class. In fact, I can remember waking up in class in the front row. I attended every class. It was a point of honor with me. Tuition in those days was $1750, a lot of money, so I made it a point of honor to attend every class. Apparently, that has gone out of style, attending every class, but I attended every one, slept through most of them.

Shustek: You were making money from the full-time job; were you also getting a scholarship or fellowship?

Metcalfe: My scholarship was $300 a year.

Shustek: How had you learned to program? You said you were programming at Raytheon. Where did that come from?

Metcalfe: I had a fraternity brother. Fraternities are a really good thing at MIT. Well, they probably are at many places. You come from home, you come to a strange city and you fall into a place that cares about you. I had big brothers. I memorized all of their names, and middle initials, and what town they're from and all that stuff as part of the initiation. John Hatcher Turner, who currently still works at MIT, was I think a junior in my freshman year. His advice was this: take 6251 and I can get you a job as a computer programmer. I took 641, which was Introduction to Computer Programming, and then I took 6251 from a famous MIT professor named John Donovan, where they taught us how to program a [IBM] 7094 scientific computer. Then I showed up at Raytheon and they said, "What's your background?" I said, "I took 6251." They said, "You got a job. Now just learn how to program this thing," and they gave me a navy submarine targeting computer to write programs for.

Shustek: In what language were you writing programs in those days – was it Fortran?

Metcalfe: Assembly language.

Shustek: Assembly language. And the Raytheon machines, of course, were totally different from the 7094.

Metcalfe: Yeah, it was the Honeywell 516-- no 1215. Anyway, it was a refrigerator-size thing that they put in submarines to do target tracking. The first program I ever wrote for money was a code converter that went from 6-bit code to 5-bit Baudot codes, back and forth, for some reason. There were teletypes involved, and Baudot codes involved. That was my first. Then later, I went on to graphical presentations of targets that submarines might encounter under water.

Shustek: What were you being paid for this job?

Metcalfe: Well above minimum wage. I was very well paid. I believe it was like $2 or $3 an hour. A lot of money. I worked 40 hours a week, so I was rolling in it. I had my own little Volkswagen. This didn't make me very popular, but I performed the following calculations. I'm working my tail off to make this money and I don't have any time, so what's my time worth? That kind of calculation. I calculated that it was not worth my time to do my own laundry. So I got a fraternity brother to do my laundry, and I paid him well to do it. His name was Stephen. He was the outstanding freshman of the class of '69, straight A's, and I was paying him to do my laundry for me. My fraternity brothers didn't like the feel of that, and I can sort of understand. But on the other hand, he was happy. I was paying him a lot of money to do my laundry. One of the other things is I decided that I would wear a clean Brooks Brothers button-down shirt every single day, because I was going to work every day. His job was to be sure that my laundry was clean, but also that I had a fresh Brooks Brothers button-down broadcloth shirt to wear every day, a clean one. Bill Stephen. He was also the New England epee champion. So here is a guy, straight A's, and we had a great relationship. It was just the other guys in the house [who] thought that there was a little something wrong with that. I didn't see anything wrong with it.

Shustek: Obviously, he had done a different calculation for what his time was worth.

Metcalfe: Yeah, and it was easy for him to do while he was doing his own laundry. He saw my problem, because I was doing a lot of things. I didn't have a spare moment. I was very efficient with my time.

Shustek: Did you enjoy programming? Is that something that you thought you might want to do as a career?

Metcalfe: Sure.

Shustek: What did you like about it?

Metcalfe: Hmm, what did I like about it…. There were the crass things about it. It paid much better than waiting tables. It was conducted in air conditioned rooms, which was a big difference in June, July and August here in Boston. But then there was the mastery of it, getting these big expensive things to do what we wanted them to do. It was like a puzzle. There was the puzzle aspect, like figuring out the crossword puzzle.

Shustek: Had you been a puzzle player and a game player? Are you now?

Metcalfe: I do crossword puzzles, but I'm not a champion or anything. I was a member of the chess club in Bayshore High School, and the math team. So I liked puzzles. Math puzzles are my hobby now. I recently wrote a program that does Sudoku puzzles. My wife, Robin, is much better at doing individual ones, but I have solved them all. She doesn't understand that. She doesn't admire it as much as I do. Robin, you solve them individually. I have solved all of them. I work with Mathematica and I do various puzzles that interest me.

Shustek: Going back to MIT, as you got out of your freshman year, you had some choices about what courses to take, and eventually, what sort of degree to get. Which direction did you head and why?

Metcalfe: I entered MIT having freshly taken art and mechanical drawing with the intention of being an architect. So my first major was architecture and I took 401, four meaning architecture, 01 meaning the first course. In the first week, the first assignment was draw your own hand. So I dropped the course in one week flat, when I realized I wasn't cut out for that and went into number theory. That lasted a year, and then I switched into physics. And then from physics, I then entered a special program my junior year at MIT in which there were 12 members called the Undergraduate Systems Program in the Sloan School of Management. So I became a management major. And then after almost finishing that, my father said to me, "We didn't send you to MIT to study management." So I stayed a fifth year and got a degree in electrical engineering and computer science. So I have two bachelor's degrees, one in electrical engineering and one in industrial management.

Shustek: It seems a little strange that with your early interest in computers and electrical engineering, you would have made a transition from architecture, to number theory, to physics, to management, before ending up in electrical engineering and computer science.

Metcalfe: Well I was taking computers and working in computers the whole time. I had a full-time job my sophomore year, my junior year, so I was programming computers the whole time. At Sloan, my interest was in the computer side of things: operations research, and modeling, and systems dynamics; the computer side of management. Then when I went to Harvard in applied math, it was the computer side of applied math, and then I finally got my PhD in computer science.

Shustek: What were your grades like at MIT?

Metcalfe: My cum after five years was 4.6 out of 5, and that, of course, includes two years in which I received no grades. This undergraduate systems program that I mentioned was unusual in that the 12 of us were exempt. We didn't have to take any courses, and we got no grades. We just received a degree at the end of the fourth year, if we wanted it. So the 4.6 was after my fifth year, and I got 5.0's in my fifth year. It was fun by then. By then, I was having fun at MIT. The best courses had arrived in the fifth year. For example, 605, Linear Systems Theory, was the best course I ever took. I loved every minute of it. And then I started learning probability and control theory in my fifth year. Like in high school I stayed that extra year and really got a lot out of it, I stayed a fifth year at MIT before getting my bachelor's degree. I got two bachelor's degrees, which is a little strange. But it was a really fun year, because I took courses I really loved, really got into.

Shustek: MIT is filled with famous legendary professors. Who were the professors there that you interacted with the most? Were any of them mentors or role models for you?

Metcalfe: I have a lot of mentors and role models. I don't really settle on one. I can give you a long list. Jay Forrester, who invented core memory, was my advisor for my junior and senior years. [Alvin] Drake taught me probability. He recently passed on. Minsky was my undergraduate thesis advisor, Marvin Minsky. I saw him twice. Once [when] I got him to agree to be my thesis advisor, and the second time, he accepted my thesis. Which, by the way, was entitled "A Neuron Model and Some of Its Information and Processing Capabilities." Fortunately, I don't think any copies of that thesis exist anymore. I remember vividly writing it on the bus returning from away matches of the MIT tennis team. I was captain of the MIT tennis team and traveled a lot between jobs and everything else. I wrote the thesis on the bus, and that's how good it was.

Shustek: So you saw Minsky twice. Were you able to get much face time with people like Forrester and Drake, or were they distant images at the front of the classroom?

Metcalfe: A lot of time with Professor Forrester. He's a very conscientious professor in terms of time with students. On that point, he invited the members of his undergraduate systems program to come to his office on Fridays for a writing class. We showed up and he asked us to write, in our first Friday afternoon meeting… Not all of us showed up, just a few, including me. He asked us to write an essay, on any topic we wanted. We wrote our essays. We came in the next Friday and we read them to each other and edited them. And he sent us home with a new assignment, which was to write the same essay. We did this, my recollection is, 19 times. The most important thing I learned from that was that Jay Forrester thinks that writing is important. So I believe writing is important, and he taught me that. Alvin Drake was a great teacher. I didn't spend a lot of time one-on-one with him; I was just taking Introduction to Probability Theory from him. I took a course from Marvin Minsky in which he was a lecturer. Artificial Intelligence, obviously, a course I took from him. J.C.R. Licklider was a professor at MIT who I actually took a job with eventually. My full-time job switched to MIT. When I graduated from MIT and went to Harvard, I then kept working, but my job was back at MIT in Licklider's laboratory.

Shustek: One tends to form very close friendships with other students as an undergraduate. Are there students that you formed friendships with that lasted or that last until today?

Metcalfe: Sure. A lot of fraternity brothers, normally, as you'd expect. I have several friends from this undergraduate systems program, like Nick Cavata[ph?], class of '68, Clyde Reddick[ph?], class of '68. I just saw him again last week. Kevin Consella[ph?], fraternity brother, class of '67. It's funny; I'm on the board of two companies at which there are principals -- one the CEO and one the VP of engineering -- who were MIT class of '68. I only met them recently. I didn't know them. The class of "68 had a thousand members in it, and apparently I didn't meet that many since I was somewhere else most of the time.

Shustek: So after five years you graduated with two bachelor degrees from MIT. Why did you go to Harvard?

Metcalfe: It was in town. It was not MIT. The advice I got was: you've been at MIT five years, you should try something else. In retrospect, I believe that was a mistake. Going to Harvard was a mistake for me. I didn't like Harvard. I hated Harvard. Did I use the word hate? I hated Harvard. I hate Harvard. I hated it. It was oil and water. It's a really long story. I just hated it from the very first day, and it's probably just bad luck. It was probably just a few rotten eggs that I ran into there that just set me off. But basically, the Arpanet -- Internet 1.0, as I like to call it -- was just getting started as a thing that graduate students got funded to do. That's how I thought you chose your PhD dissertation. You say, "What's ARPA funding now? Arpanet, okay, I'm going to do that. So here I am at Harvard, a first year grad student in the PhD program in the division of engineering and applied physics. I had just taken a full year of digital design projects at MIT. I actually helped build part of a computer, got really facile with digital electronics. What we were busy doing, all of us across the universities around the United States, was connecting our university computers into the packet switches, the IMPs of the Arpanet. And there's Harvard, with a PDP-10 and an IMP. So I said, "Hey, I just graduated from MIT learning how to program digitally designed things. Why don't I design the IMP interface to connect the PDP-10 to the IMP?" And Harvard said, "That's too important for a graduate student to do. We're going to get a company to do it." Subsequently, they hired a company that you may have heard of called Bolt, Beranek and Newman, the very designers of the IMP, to do it. BBN then turned around and hired another graduate student to do it for them-- Ben Barker is his name-- which I resented. So I turned around and went down the street to Project MAC, now known as CSAIL, at MIT. I knew some people there and there were some openings. I said, "You have an IMP and you have a PDP-6," and then a PDP-10 later. "I'd like to put those together, since I have just taken a couple of courses." Al Vezza, who was part of J.C.R. Lickider's lab said, "Okay, we'll give you a job. Your job is to connect our IMP to our PDP-6. And remember, it's going to be a PDP-10 later, so you have to anticipate that." So he gave me the job. I started designing. So here I am again, a student at Harvard, but a full-time employee at MIT, which by the way, pays better. I was paid more as a member of the research staff at Project MAC than my advisor at Harvard, which annoyed him to no end. He thought I should be working for slave labor wages as a grad student, but I had this cool job. I turned to Harvard and I said, "You know I'm designing this interface. I could make two copies. I could give you one for free." That's when they reminded me that Bolt, Beranek and Newman was going to build their IMP interface and they didn't need mine. Anyway, that got me started in the high-speed network interface business, designing this. I have it upstairs, by the way.

Shustek: How did those interfaces turn out, yours and the one done by BBN?

Metcalfe: They both worked. Mine took me a year. I built it and then it sat there for 13 years. Then it was decommissioned and they sent it to me in a Federal Express package.

Shustek: Did you do the software as part of the PDP-6/-10 to drive it, as well?

Metcalfe: Yes. I did most of it. My buddy there was a guy named Bob Bressler. He did the software coming down and I did the software coming up from my hardware. Then we split the middle. So we put the PDP-6 on the Arpanet, Bressler and I, and then I started writing my Harvard PhD dissertation on that topic. This is another reason why I hate Harvard. I, in my mind, was slated to receive my PhD in June of ’72. That is two years after my master’s degree. So I drafted this thesis and went job hunting. The committee was up at Harvard. I recruited a committee, like you’re supposed to recruit. I submitted my draft there, but I didn’t pay really close enough attention to what was going on at Harvard. I went out job hunting and I got nine job offers, all predicated on the fact that I would be receiving my PhD in June of ’72. Well on May 19th, or 20th, or 28th, or some really late time, I went in to do my thesis defense. I came into a room full of people who I vaguely recognized and promptly failed my defense, which is not a common event. Although I do share it; the current head of the National Academy of Sciences, Bruce Alberts, he also failed his from the same department at Harvard, so we had that in common.

Shustek: Who was your thesis advisor?

Metcalfe: It changed. I’d rather not name the one under whom I failed, but I succeeded a year later under a man named Jeff Buzen, who is a local entrepreneur of some renown, later went on to found BGS Systems and was a fantastic guy. I knew my relationship with Jeff was going to go well when in our first meeting, he said, “Okay, what do we have to do to get you out of here?” I said yes, that’s the right attitude. And why did I say that? Because I had these nine job offers. And I had nine job offers because I had been working on the Arpanet, which was ramping. All these places out there that hang around ARPA thought it might be a good idea to have someone work there who was an ARPA guy, so it was easy to get nine job offers, because I was in the mainstream of funding of ARPA.

Shustek: What sort of places, research labs, engineering companies?

Metcalfe: Let me name them; MIT, although had they offered me a job of assistant professor at MIT, I would have taken it, but they didn’t. As you’ll learn in a few seconds, I took another job offer. I was offered a professorship at the University of Kansas and I would have been so happy had I gone there. It was a fantastic place. I went to a basketball game there and the stadium was packed. I loved the professors. Lawrence, Kansas is a beautiful little place. What would my life have been like had I gone there? I don’t know.

Shustek: Very different from Boston and New York.

Metcalfe: And I got an offer from SRI (Doug Englebart), Xerox PARC, BBN and a few others, but I ultimately chose to go to Xerox PARC, which was just getting started.

Shustek: We’ll talk in a while about why you went to PARC and what happened when you were there, but let’s go back to some of your experiences with the early Arpanet hardware and software. What sort of Arpanet-related projects had you worked on at MIT?

Metcalfe: Well, as an undergraduate I took courses in digital design. In particular, this is relevant. I took the advanced digital design lab course where we built a computer, and my job in that project was to build the main memory. The way I built the main memory was with an acoustic delay line, which meant I had to build this little digital circuit that would launch these bits down this really long cable, acoustically carry the bits around in circles like this, and then they finally came out the end and then they came back into a digital circuit where I had to recover them and then send them back in. So it was a delay, a dynamic memory that you had to constantly refresh, but of course it was a transmission line. So the whole idea of transmitting and receiving was built into that problem. Then when it came time to hook the Arpanet packet switch called the IMP -- the Interface Message Processor -- to the PDP-6, once again I was launching circuits along this big cable between these two boxes. It was a struggle to build this piece of hardware which is about this big. I have it upstairs. It's four small boards all wire wrapped with the chips jammed in these sockets and then wire wrapped and the cable coming in and then a bunch of lights to help with debugging and knowing the status of the thing. It was really challenging, especially for a first time design, to get it to work.

Shustek: What was the technology? Was this 7400 series SSI and MSI?

Metcalfe: 7400 SSI, yep. My favorite was the 7406 with hex inverters, and I could fix lots of bugs with that just by introducing delays by N inverters in a row. I was later told that was bad design, but it worked for me and that device worked for 13 years, I might add. By the way, it was for a PDP-6, and the Internet protocols had not been developed yet. That is, when I was building this hardware we had… In fact, Danny Cohen at Harvard and I devised an INCP, an Interim Network Control Program, that was later replaced by the NCP, that was later replaced by TCP/IP more than 10 years later. So we were dealing with the raw packets here. There were no protocols. We invented all the protocols ad hoc for our various projects. One of the things that was up in the air was the word length. Now in those days, you had 18 bit machines, and 36 bit machines, and 32 bit machines were kind of a new thing, and it went on. So it wasn't clear what the word lengths were going to be or what the patterns should be in memory so in my IMP interface you specified the word length you wanted and this baby would assemble that many bits and then jam it into the memory of the PDP-6. It was kind of a funny thing. So this device is quite big because it has all this modularity in it and flexibility, all of which I designed out in subsequent high speed interfaces.

Shustek: Is it all hard-wired logic, or was there any microcoding state machine type stuff?

Metcalfe: All hard-wired. No state machines. I learned about state machines at Xerox PARC, actually. But anyway, I designed this thing. It took me forever, a good portion of a year. I finally had it, and I got it working with my test programs. It would work, but it only worked for 15 minutes and then it would go "bah", and all the lights would go in the wrong direction and then it would die. Then I would reboot and then it would run again. I had these extensive diagnostics, random patterns and everything, to just test the hell out of the thing, different word lengths, test every edge case. Then it would run just fine, over and over again through every- and then it would just die. I worked on it for a month and I couldn't figure it out. So I picked it up, and I walked down the hall. This is the ninth floor of Technology Square, so I was in Licklider's lab and the AI lab was at the other end of the floor, and Multics was on the other side of the floor. I went down to the other end where one of my buddies and geniuses named Tom Knight, who is still at MIT. He built hardware, wrote computer programs, and had been there ahead of me. I walked up to Tom and said, "Tom, I need your help. I cannot get this thing to work and here's all the drawings and everything. Would you be willing to just sit down and take a look at this thing?" And Tom looked at it and he asked me a few questions and then he said, "Well, I know what's wrong." "Really, Tom. Pray tell." He says, “Well, you don’t have any bypass capacitors on it. Now I know you just took a bunch of courses in digital electronics but at some point everything is analog. So what’s happening, Bob, is when certain patterns get into your registers and they all go 1 and all the transistors turn on, they take too much current, too many electrons, and then the voltages start to droop because you’ve sucked in all those electrons and then all the digital devices start to malfunction. So what you need to do, Bob, is sprinkle some bypass capacitors here and there to store up some extra charge for those cases when you have lots of 1’s in your registers.” So I say, “Tom, where should I put them and how many should I put?” And he says, “It doesn’t matter very much. Just sprinkle them around.” So I went quickly back to the lab and got my soldering iron and I soldered on -- I probably put a bypass capacitor on every third socket to this huge board. And I plugged the sucker in and she worked for the next 13 years. Anyway, I learned a lesson there -- the analog digital -- which came in handy later because Ethernet is a combination of analog and digital itself.

Shustek: The nasty realities of digital design.

Metcalfe: Right, and I’ll always remember Tom Knight. He knew. Knowledge is power and he knew in an instant. Just by looking at the board he could tell.

Shustek: You mentioned having worked with INCP with Danny Cohen. Did you also work on an aircraft simulator with Danny?

Metcalfe: Exactly. This is a funny story. Steve Crocker, who worked at ARPA at the time, he and Jon Postel, another famous Arpanet guy, they started a thing called RFCs -- Request For Comments -- whereby people working on the Arpanet and ARPA contractors would write memos about this and that, and then publish them in this comment series. So Danny Cohen and I -- he at Harvard, because I was at Harvard too, sort of, a student, but I was working at MIT -- we decided to work together. We had just finished, BBN had attached Harvard’s PDP 1 and PDP 10 to the Arpanet, and I had connected our PDP-6. We had an Evans and Sutherland image processor, clipper-divider or whatever it’s called, on our PDP 6/10, and he had an aircraft simulator running on the PDP-1. So we devised the following experiment, where the simulator would run on the PDP-10 at Harvard and it would send its images across the Arpanet to my PDP-10; say it was a -10 by then. Then we would give them to the Evans and Sutherland image processor that would do the clipping and the rotations and the dividing, and produce a return image which then we would send back over the Arpanet -- that is, across town to Harvard -- which then he would receive on the PDP-1 and then it would display it on the PDP-1. We had to write all the low level packet protocols for getting this to happen, and we got it to work, and it was kind of cool. You could imagine better ways to do it, but it was a stunt. I wrote an RFC about it, and this gives you insight into an obnoxious side of my personality. The name of the RFC was “Historic Moments in Networking”, in which I wrote about this experiment. I’d actually thought it was an historic moment in networking that everyone should know about, and so we published it. I forget what its number was. Maybe it was RFC 89. It was—

Shustek: Exactly right.

Metcalfe: --a two digit RFC. I wrote a few more of those, got into the three digits after a while.

Shustek: Don’t you think in retrospect that it was an historic moment?

Metcalfe: Yeah, but you’re not supposed to be so self-possessed to write that at the time. Other people are supposed to call it historic. You’re not supposed to call it historic. But there is a little bit of a promoter in me, I guess. What this illustrated, though, is we wrote an INCP, an Interim Network Control Program, to get this to work. We wrote programs for a long time to get this to work, and then threw them away. That is there was nothing-- This was just prior to the NCP that we -- others principally, but I among them -- came up with. So a protocol standard was developed, called NCP, for how you would reliably send packets through this Arpanet switching thingy and recover them. Then we invented a protocol, which maybe you’ve heard of, called the Telnet protocol for doing remote logins. We documented that in some RFCs and I implemented it for our PDP 10. Oh, by the way, each of these PDP-10s ran a different operating system. My PDP-10 ran ITS [Incompatible Timesharing System], which is the same one the Artificial Intelligence lab ran. BBN had another operating system for the PDP-10 called Tenex. So we had to do the Telnet protocol over and over again for each of these operating systems.

Shustek: I checked the index of the RFCs, at least the first 20 years, and I think you had contributed to nine of them altogether. One was interesting. It was RFC 602, which was a warning that you issued, this must have been ’71 or ’72, about security faults in the Internet. The RFC was called “The Stockings were Hung by the Chimney with Care”. Do you remember that one, and the warning you were issuing?

Metcalfe: Yes. I’m very proud of that one because I claim, when I’m bragging, that that was probably the first time anyone warned the Internet community that we were not paying attention to security. What had happened was a couple of high school students hacked into the Arpanet through one of the early TIPs. A TIP [Terminal IMP] was a way to dial into a Telnet program so you could then log in to any of the computers on the Arpanet. Imagine that! And some high school students, as I recall they were probably in Los Angeles around UCLA, somewhere like that, managed to find out the phone number. No one was keeping it a secret. Acoustically coupled modems were the big thing in those days, and so these high school kids got in to some of the computers and did some mischief. They caught my attention, so I wrote this 602 RFC saying “Look out! Trouble!” Of course we still have security problems many, many, many years later and we will forever.

Shustek: It is now 33 years later. Do you think the problem has gotten better or worse? Do we have a handle on the security issues now?

Metcalfe: We’ve made a lot of progress. Security will never be solved, ever solved. It’ll always be a problem because it’s a tradeoff, constantly a tradeoff. We’re a lot better at it than we were. The problem was we were grad students designing the Arpanet, and there were lots of things unimportant to us. Our goal was to get connected. We didn’t care about security, because who would threaten it? We were a small family then, so we didn’t really anticipate that anyone would be malicious. The other thing we left out was economics. Since we were being paid, we just didn’t put any economics in the Arpanet. Eventually that was overcome in the ‘90s with the development of economics -- like advertising for example. But still, the Internet is recovering from the lack of early attention to security and economics.

Shustek: In October of 1972 there was a very famous conference at which the Arpanet and its interconnectivity was demonstrated. You had a role in that conference. Can you describe what you did?

Metcalfe: Yeah. In ’72 the ARPA people, led principally by Bob Kahn, decided to have a “coming out” party for the Arpanet. They scheduled it, I remember vividly, in October of ’70 at the Hilton Hotel in Washington, D.C., where we were to have a two or three day demonstration of the Arpanet. And I’m out in Palo Alto, California, working. By then I had gone to-- Wait. Seventy?

Shustek: Seventy two.

Metcalfe: I had just gone to Xerox PARC, in June of ’72, which is when I was supposed to get my Ph D. from Harvard but didn’t. Xerox said, “Come anyway and finish while you’re here,” thank God. When I caught wind of this demo -- I was invited to participate in this demo -- I said I know what I will do. I will write the book that people can use to wander around the Arpanet and get various demonstrations. So I wrote the book. It was a booklet, and it was called “Scenarios for Using the ARPA Computer Network”. All it was was a manual for what you type: when you go here, and what you go here, and what you do when you go here and here and here. Then I went to the event. Now this is 1972, so imagine me wearing a big red beard -- huge, red beard -- and wearing wing tips. I loved wing tips because I had been to the Sloan School of Management as sort of a semi hippie, a right wing hippie. So I’m helping give this demo. They gave me packs of visitors to give the tour to, which meant going around from teletype to teletype and then giving them copies of the book that I had edited, this booklet. They gave me 10 AT&T executives to give the tour to, all of them wearing three-piece pinstriped suits. I started on the tour, and was very proud of this project and very enthusiastic about it. Right in the middle of the tour, the packet switch, which was on a pedestal in the middle of this Hilton Hotel ballroom so that everyone could see it, crashed. The one time during this two or three day event -- I forget how long it was really -- it crashed, while I’m typing. Imagine me. I’m sitting there. There were 10 guys in pinstriped suits behind me and I’m typing, and “Oops.” You know how in those days you’d hit the teletype keys and nothing would happen; the Echoplex wouldn’t plex. You’re going like this and you get that dead sound that a key makes when nothing’s happening and oh, it was really bad. So I turned around to begin to explain and apologize, and see behind me 10 people all smiling broadly. The picture is burnt into my [memory]. These people were amused and pleased that the Arpanet had crashed, because this confirmed for them that they were going to make it to retirement without having to try to deal with this technology. We were threatening the telephone company. We were using their circuits. They had these very high speed circuits, 50 kilobits per second circuits, and we were using them to be the backbone of the Arpanet. But we were talking packet switching, and we were being nasty about circuit switching and how old fashioned it was, and they didn’t like that and they were happy. I’ve had this thing about AT&T ever since. Of course, AT&T was subsequently in 1984 gutted, but it’s coming back now so I have to be careful. I’m going to have to go into the basement and rearm now that AT&T is coming back.

Shustek: This was 1972. Was there email yet? When did you send your first email and were you part of the Message Group community and all of that?

Metcalfe: Well, we had emails in the ‘60s but nothing like… We had no networks in the ‘60s, but we had email. Within a timesharing system you would have e-mail. I remember Larry Roberts, who was arguably the father, or one of the fathers, of the Arpanet and the Internet. He's largely been forgotten now, but he was the man in those days. So here's a guy, he's head of the ARPA Information Processing Techniques office, the funding agency that had funded the whole Arpanet under his direction, and this man had a TECO macro for writing and reading email.

Shustek: TECO was a text editor.

Metcalfe: TECO, text editor, a programmable text editor with macros. We all used TECO then. This was before everything. This is a text editor in which you could write computer programs in it for manipulating your text. Larry had written some TECO macros for formatting e-mails and showing the columns and the rows and the To's and the From's. I thought it was really cool that the head guy who had administrated all of Arpanet was writing TECO macros. I remember that was really cool. That's when the famous at-sign invention [happened]. Ray Tomlinson, who was at that time privileged to be a system programmer at BBN working on the Tenex operating system, which became the preferred operating system for PDP 10's, wrote the… Tenex had been proliferating out of BBN to all the ARPA sites. His email program was the one that was proliferating out of BBN. So he became the inventor of email, and got to be there when at-sign was chosen to separate the name of the person from the name of the site where the email was done. We all adopted it. He didn't really invent email, but he made this pivotal contribution of this email program for Tenex with the at-sign, and the rest is history.

Shustek: This was before TCP, before DNS. How did you address hosts on the network?

Metcalfe: Interesting that you should ask. One of my early contributions to the Arpanet was to organize a conference at MIT called the System Programmers Workshop that Al Vezza and I… Al was my boss, since retired, at MIT. We organized this and I was the system programmer for the Arpanet, Bob Bressler and I. We organized this System Programmers Workshop and invited all like people to come to MIT. What we were going to do was test the connection of every machine to every other machine in an N by N matrix, and we were going to log in to them. I wrote a survey program. I wrote a program that would just systemically go and try to log in to every other site on the net, and display how successful it was: how long it took, whether it was a successful connection, what happened. It was called "Survey", that I wrote. I had to have a way to refer to these machines and there was no DNS and there was no… So I just made a little text file, and it contained the octal of the host number and the text that would be displayed when that octal was encountered in the correct field. The survey would then say "MIT DMS", "MIT AI", "MIT Multics", "Harvard PDP-10", "UC Santa Barbara 360/91, "UCLA Sigma 7", "SRI" -- names that I typed into a file. So then when we did the N by N matrix of all these connections we just went to my little file, and every time we came up with a little octal number we would just look up the text and display it. This is way before DNS, and of course it made sense that this should be. Every system programmer had such a file. Right? They all had to do this over and over again. Somewhere along the line someone said, "Duh. Why don't we have a system for automatically updating these files and sharing them?" And that became the DNS. I didn't do the DNS but someone did at that point.

Shustek: Was this before SRI had the Network Information Center, the NIC?

Metcalfe: No, no. The NIC existed in ’72. That booklet I wrote for the demo in October of ’72 I wrote at Doug Engelbart’s lab at SRI on Ravenswood Boulevard. I’d drive over there and type it in to NLS [oNLine System] with the mouse and the key set and the little.. The book was published at the -- what was it called? -- the NIC, the Network Information Center at SRI, at the NLS, the Online System, NLS, that Doug had developed.

Shustek: You, at that point, had already pointed out that we would have security problems with these networks. Did you also anticipate problems with email, spam, authentication, flaming, that sort of thing, that developed later or were these days when everything was benign?

Metcalfe: We were innocent. There was certainly no notion of spam, because advertising didn’t come on to the Internet until the ‘90s I’m guessing. No. There may have been some spam in the USENIX user groups. There may have been. I wasn’t in that community, so I don’t remember. But no, in the Arpanet days that we’re now talking about I don’t remember any problems like spam or viruses. The first virus I knew about was at Xerox PARC, later in the late ‘70s. John Shoch there did the worm thing. That’s the first I knew about a virus.

Shustek: Let’s take a short diversion here. The Arpanet that you had worked on eventually becomes the Internet and the World Wide Web and is obviously something that’s changing all of our lives. I think I remember correctly reading that you politically tend toward libertarianism, the idea that small government is best. Yet all of this early networking work with ARPA was funded by the government. Do you think in retrospect that that’s a proper role for government? Should they have done that and if they didn’t would anyone else have done that?

Metcalfe: No, I think they should have. I think one of the few things government should do is finance research. I have learned, from many years, that the only companies that can afford to do research are monopolies. Real companies can’t afford to do research other than monopolies. There’s some famous ones, like the telephone monopoly, [AT&T] Bell Labs; the computer monopoly, [IBM] Watson Labs; the copier monopoly, Xerox PARC. And on it goes. In retrospect, the monopolies aren’t worth it for the research they do. It’s nauseating how much we hear about how cool Bell Labs is, or was. But other than the transistor, UNIX, and the Princess telephone, what did we get for all that money? And then for years AT&T as a monopoly sat on innovation, and IBM after that, and Xerox after that. It’s just not worth it. So let’s kill those monopolies and if we need research have it done at research universities. The other spin I would offer there, as a practitioner of technological innovation we worry about technology transfer: how do you get technology transferred from the lab into the marketplace? The best way to do that is with people, and it is the business of universities to graduate people. So let’s do our research there. I think the Arpanet is a great example where the government financed some research. Now the Internet that was built… this is a sore point. There are people around who think that the government paid to build the Internet. That’s not even roughly true. There were some early developments, and some protocols designed, and some startup funds like CSNET. But the Internet was built by 3Com Corporation, and Sun Microsystems, and Cisco, and huge companies, huge efforts, in which there was just a little [??]. The amount of money that the government put into that research was pitiful. ARPA in my day had an annual budget of $40 million a year, compared to the billions -- we were very angry about this -- there was small numbers of tens of millions of dollars going into the Information Processing Techniques Office and there were billions going into fusion. For our tens of millions we have given you the Internet, and for those billions where is the fusion?

Shustek: Do you think the level of government funding these days is appropriate?

Metcalfe: That's very complicated. I know it's up phenomenally over the last few years, so if it's not enough it's certainly better than it was a few years ago. There's a lot of whining out of the scientific community about funds being cut, and this and that, but you look at the numbers they're up and up and up. The reason that there's whining, of course, is someone has to decide about priorities. And the priorities of late have been to the NIH instead of to the physical scientists. So all the physical scientists are saying, "They're cutting the R&D budgets." Yeah, well, sort of. They're cutting your R&D budgets, but look at NIH. That budget's been doubled in the last few years. But I don't know if it's appropriate. I do know that there are people whose job it is to say more would be appropriate, but that's not really an answer. That's just called lobbying, and the National Research Council and the academies are very actively lobbying for increases of expenditure. But how much is appropriate? That has something to do with national priorities. It also has something to do with how many scientists are there competent to make good use of that money? Are there enough of them, or are we just dumping our dollars into places that won't yield results? I don't know how to answer that. That's very complicated.

Shustek: Let's return to your career. You're leaving Harvard with a Ph D thesis that was rejected, and you got nine job offers. Why did you decide to go to Xerox PARC?

Metcalfe: I went to Xerox PARC because they offered me more money than anyone else. It was, I believe, $19,000 a year, something like that, not 20. Xerox was a booming monopoly in those days, and everybody flew first class. First class was reasonably priced in those days. The research center was new and spiffy, and beginning to be full of really interesting people like Butler Lampson and Alan Kay. Bob Taylor and Jerry Elkind, who were sort of the leaders of the computer science laboratory there, told me they really wanted me, and I wouldn't have to teach, and I could do whatever I wanted, and I was going to be the network guy. So I said yes. Oh, and I'd always wanted to go to California. During the '60s there was a group called the Beach Boys, and they touted the California life. I remember "I Wish They Could All be California Girls". Be on my woody, surfing, California… Little did I know that Palo Alto was really far from the ocean, and the beaches there -- the water's really cold, but… So I moved to California. That was another factor: just to see what California would be like.

Shustek: You didn't understand the distinction between northern California and southern California?

Metcalfe: I did not understand that distinction. To me LA and San Francisco were the twin cities area. By the way, there was another thing. In those days Route 128 was the entrepreneurship capital of the world and I was in the thick of it. I hadn't noticed Silicon-- there was no Silicon Valley that I knew about in those days. That was '72. Intel would have been 4 years old at that time, so just getting started. There was no Apple. There was an HP, but they barely made computers, just barely. I'm not even sure they did. They did later. Yes, that's right, because when I taught at Stanford in '75 I taught on an HP mini computer. But still HP was smaller than Xerox in those days. Old and respectable, but a tiny little company by comparison. So I was leaving this hotbed of DEC, Data General… Wang, I guess. When did they get founded? Well, DEC certainly was around by then. Data General was too. Let me see. I don't know about Wang. Anyway, this was the hot area and I was leaving it to go to the beach, basically.

Shustek: So you arrived in Palo Alto. Who did you work for and what was your first assignment?

Metcalfe: Well, I joined the Computer Science Laboratory. My bosses were Jerry Elkind, who had been at BBN and a year at Sloan, which is how I met him, and then he became co-head of the Computer Science Laboratory. Bob Taylor, who had been at ARPA, was the co-head of it. They were my bosses, but I was in there with Butler Lampson and Alan Kay and Chuck Thacker and Ed McCreight and a long list of other people. My job was to be the networking guy.

Shustek: For which machine?

Metcalfe: Ah ha. There was a funny thing. Xerox had just then acquired Scientific Data Systems, a hot minicomputer company of the day, whose arch enemy was DEC, which made the PDP-10. But we were all ARPA guys and we liked the PDP-10. We wanted to have a PDP-10. Xerox wouldn't let us buy a PDP-10 because it had just bought SDS, so we built our own PDP-10. My first project there was to join the effort to build what was called the Multiple Access Xerox Computer, or MAX, or MAX-C it was sometimes pronounced. My initial job was to write the floating point microcode to emulate the buggy floating point operations of the PDP-10. We had a goal that this machine when we built it had to run the DEC diagnostics unmodified. Eventually Ed Fiala did all this, but I got started doing the microcode and we wrote floating point for the PDP-10. Then we ran the diagnostics and realized that the PDP-10 had bugs in it. So we had to put those bugs into our microcode so we could run the diagnostics. I just got started in that project and then Ed Fiala took it over, because they had other work for me, which was to put this PDP-10 on the Internet, something I had done just recently at MIT. So I set about building an IMP interface to connect this fake, ersatz, clone PDP-10 to the Internet. There was an IMP there; Xerox PARC got an ARPA contract whose sole purpose was to give Xerox PARC access to the Arpanet, which was then just for government contractors.

Shustek: Why was Xerox building the MAXC? Was that going to be a product?

Metcalfe: No, a research machine. That's a good question. I think we were just building it because we wanted to, and it would be a tool for the scientists there. We couldn't have our own PDP-10 so we just built it. But in the course of building it we developed a team institutional knowledge of how to build a computer. Thacker and Lampson had done it before; they'd been involved in building previous computers. We as a team learned how to build a computer. Right after we finished MAXC we started work on the Alto computer, arguably the world's first personal computer.

Shustek: What was the environment like at Xerox PARC? Was it similar to what you had experienced at MIT? Was it different?

Metcalfe: Completely different. It was very laid back. There was only one meeting per week. It was called…god, I've forgotten. It had a funny name. Anyway, it was every Tuesday at noon.

Shustek: **"**Dealer".

Metcalfe: **"**Dealer" it was called. The only meeting that you really had to go to was Dealer. This created problems because Dealer was at a fixed time. I forget when it was, but say Tuesday at noon. The way I lived my life then was I would work until I got tired, and then I would go home and go to bed and then I would sleep until I woke up with no alarm clocks. Then I would go to work and stay there until I got tired. So my day precessed. I was frequently there all night. But then Tuesday would come around and I had to be there at noon. So then I had to get resynchronized to go to Dealer. You wouldn't want to miss Dealer both because it was expected you'd be there, but also because it was fun. We'd sit famously in these beanbag chairs, and we would have speakers. A speaker would come every week. Job talks usually. Then we would discuss what was going on at the lab. It was heaven on earth, because we had all the tools we needed. Money was not a problem. If you wanted to buy an oscilloscope, you got an oscilloscope. If you wanted to buy a printer, you got a printer. If you wanted to buy a modem, you got a modem. You needed something built, there were technicians there to help. And it was California, so we rode bicycles and we-- I remember we used to ride our bikes along Arastradero Road over to Alpine Road and go to Rissotti's, the Alpine Inn that you're familiar with, at lunch time. We would drink beer at lunch, and that meant the afternoon was gone and that work might resume that evening after dinner. It was a very idyllic time.

Shustek: People were collegial? There weren't prima donnas who got under the skin of other people?

Metcalfe: There was a little of that. The worst case of it involved me, because I'm the kind of person who gets under people's skin. I think there were other conflicts, but the one that's most vivid to me was between me and Chuck Thacker. Thacker and I just… He's a wonderful man and a genius and all of that, but he and I just don't get along and that's just the way it is. I don't hold it against him. I'm sure it's my fault but we had a big problem, him and me. With apologies to Chuck, who's now a big shot at Microsoft, he was the principal designer of MAXC upon which I was a minor contributor. My job was to get the MAXC put on the Arpanet. I had a card that plugged into a [Data General] Nova [minicomputer], and the Nova was connected to the MAXC. Then I had to get the NCP software, the Arpanet software, running inside of Tenex. We were running Tenex on this fake machine. Then I had to connect the NCP to my little packet switch to go out over the IMP. I had a hell of a time debugging it. At one point I thought I had all my bugs taken care of, and the damn thing still didn't work. I went to Chuck and Chuck just… He and I just never got along. There was just nothing but animosity. I think part of it was he doesn't have a PhD, and I do. Something about that made me misbehave, and made him misbehave, with respect to each other. Because he was much more senior and accomplished than I, but I had this flashy Harvard PhD. That may have been the beginning of it. But when I went to Chuck to say, "Look. I can't get this to work. Can you help me?", he blamed me. He said, not in so many words, but "Just make it work, idiot."

Shustek: He didn't have a simple answer like "add decoupling capacitors".

Metcalfe: No. But I worked hard at this. I began to notice as I'm trying to debug it -- this is very complicated stuff, lots of… whole operating systems, memory, networks, all changing, zipping around -- I began to notice something. It was like I was having a mirage. I'd look-- You're at this glass teletype and you're looking at memory locations and I began to hallucinate. I was looking at memory locations twice and they were different when I looked at them.

Metcalfe: I'd like to reiterate at this time my high regard for Chuck Thacker and what a great man he is. The fact that he and I just don't get along is my fault. It's nothing wrong with him. So, in the middle of this story. Here I am. I've been given the job to connect the ersatz PDP-10 to the ARPANET IMP so that Xerox PARC can be on the ARPANET. It's a job I just finished doing for MIT, thought I was going to do it for Xerox. So I built this card again, a simpler version of the card. My second high speed network interface, plugged into a Data General Nova 800, I think it was called, and that hooked up to MAXC. Then I had to get the Network Control Program under Tenex, which was running on the MAXC, which was just coming up. I had to get it to talk to this 800 that talked to the ARPANET, so that the Telnet and the email and everything on Tenex would work through the ARPANET. The machine is just coming up, but it's now running Tenex and there are users typing on it, but it's being debugged. At night I would go in there to get my network code running. And I'm trying to debug it and I'm having... The code's all written now, and I've debugged all little pieces of it, and I'm trying to get it to all work all the time, which is 95% of the work. And I've reached a point where, again, I cannot get the damn thing to work and I'm there terrified I'm never going to get it to work and this will be the end of my career, say. I began to hallucinate. I began to think that I was looking at memory locations and when I would look at them again they would be different, which is obviously a hallucination. I began to suspect that maybe the MAXC computer wasn't entirely debugged, and that I was running into a bug that was Chuck's bug, not my bug. So I went to Chuck to get his help with this, and he, in not so many words and I'm sure he didn't mean any harm at all, basically said it's your fault and go fix your own bug and don't bother me kid. Because this machine -- my machine -- works. It's your stuff that doesn't work, you and your fancy Ph.D. You can't get it to work, nah-nah-nah. That kind of thing that goes on among adults in a research lab. So I kept working on it, and my hallucinations became more frequent as I began to see that memory was not remembering stuff. I said, "Chuck, the memory appears not to be remembering stuff, and this makes it very hard to get your program to work when the instructions change." And Chuck said, "Look at this." He ran his memory diagnostic. Zzzzz, patterns, 0's, 1's -- it ran for hours, no problem. Flake off. Weeks are going by. My career is falling through my fingers -- what remained of it because I didn't have my PhD. Maybe I was about to get my PhD; I was in that awkward period. I decided to do the following thing. I'm a PDP-10 programmer of some renown. I thought I would write a user-mode application program that tested memory. I called it Munger, m-u-n-g-e-r, because it would mung memory. It would just, in virtual memory, in memory, right there in user mode, it would just write a pattern, and all different. I was very thorough, so there was all 0's, all 1's, alternating 0's, alternating 1's, random, the whole pattern. And if there was ever a difference detected between what had been stored and what was retrieved, the program would type Control-G, which you may remember was "bell". I got the program all written and assembled, and I typed munger, at the Tenex command line and <snaps finger> six seconds later, "ding". It starts dinging, "ding, ding, ding, ding". "Chuck", I say, "Chuck, come here, look." Of course the program, after ringing the bell, printed out what had been stored, in what location and what had come back out, and showed the differences. And there were some bits different. "Chuck, look, in virtual memory and user mode I'm getting memory errors." His first reaction was, "There's a bug in your program". Well, it turned out there was no bug in my program. It turned out that Chuck's memory diagnostics tested the CPU port into memory, but they did not test the disk port into memory. So every time this timesharing system swapped out to disk and then loaded the memory back in following a page fault, there was a chance that the page would come in different. “Oh gosh, yes Bob. You found this bug, and we are now going to fix it. We need to improve our memory diagnostics to test the port to disk”. And guess what? My NCP started running and everything became hunky dory, and it ran for years thereafter, and it all went away. But not before seriously damaging my relationship with Chuck Thacker. I kind of held this against him, and he kind of held it against me because I was so obnoxious as to write this program called munger, which I ran for years thereafter and it never got a bell ever again. To his credit, he knows how to build a computer. This very same thing happened on our next computer. After the MAXC got running, we decided – Chuck, Butler, Allan Kay decided -- to build the Alto, the world’s first personal computer, and I’m the networking guy. This very interesting thing happened. Butler Lampson took me over to see Charles Simonyi. Charles was working on a network for Xerox PARC. It was a 50 megabit per second version of the 50 kilobit per second ARPANET, with little tiny high speed packet switches that would be routing packets around the building at 50 megabits per second, as opposed to 50 kilobits. Charles called this network Signet- s-i-g-n-e-t; Simonyi’s Infinitely Glorious Network. Butler said, “Bob, you’re the networking guy, I think you should take this project over from Charles, because Charles, I’d like you to work on our text editor called Bravo.” Charles said “fine”, I said “fine”. He went, wrote Bravo, wrote Word. He’s a billionaire, to make a short version of that story. But then I got Signet and I started looking at it. I had just then gotten my Ph.D. approved. The way I got my Ph.D. approved, that opens up a whole ‘nother story. I’m going back to June of ’72, Harvard. I came back into the room after my oral defense. You know you do your oral defense, then you leave the room, then the committee meets, then you come back in and they tell that you’re going to get your degree in two weeks and that your parents are going to be proud of you. Instead they told me, “We do not think that your dissertation meets the requirements of originality, and it’s not theoretical enough; it’s simply engineering” -- which is a very Harvard thing to say -- “and we don’t think it rises to the level of a Ph.D.” Just between us, they were right, but I will never admit that to Harvard University. I then picked up the phone and I called Bob Taylor at the Computer Science Laboratory at Xerox PARC and said, “Bob I got really bad news, I just flunked my oral defense. I will not be receiving my Ph.D. next week or the week after.” Taylor said, “Well okay, well come anyway, and you can finish your thesis from here, no problem.” God Bless Bob Taylor. So I did. I go to Xerox PARC and start working on MAXC and all that. But in the back of my mind I’m working on my Ph.D. dissertation with Jeff Buesen, my new thesis advisor who was committed to getting me out. But I needed to make my dissertation more theoretical. One day I was visiting Washington, DC. I was an ARPA contractor, and I was visiting ARPA and staying at the home of Steve Crocker, who was an ARPA program manager. I stayed at his house and I slept on his couch in his living room; a foldout couch, not unlike the one in the next room. I think I bought that one to emulate Steve’s couch. But I was jetlagged, a very common phenomenon. I couldn’t go to sleep because I was on California time. There was a shelf behind this couch, and on it were a bunch of books including the AFIPS Conference Proceedings, 1970. I thought, what better way to go to sleep? So I pull out the AFIPS Conference Proceedings ’70 and I open it up and there’s a paper, “ALOHA Network” by Norm Abramson, 1970 AFIPS Conference Proceedings. And I begin to read this paper.

Shustek: You had not seen it before?

Metcalfe: Never heard of ALOHAnet, never heard of Norm Abramson, never heard of anything. This is in that period in ’72 while I’m looking to be more theoretical. I read this paper and it has math in it, which is theoretical. The math was basically a Markov Poisson probabilistic model of the performance of an ALOHA packet radio network that Norm was building at the University of Hawaii. Thanks to my courses at MIT, I understood the math. It was very simple math, it was a simple model. But there were two things about the model that annoyed me. You know how queuing theorists like their math to be tractable, so they make certain assumptions -- “we” make certain assumptions -- to make the math tractable. Norm had made two assumptions about the operation of the ALOHA packet radio network that I found annoying. One of them is, he assumed that the network had an infinite number of users. But I knew better. I knew the ALOHA network, from the paper, had three users or nine users, but it didn’t have an infinite number. The other thing he assumed, was that these people who were basically typing on terminals connected by radios to a 360 mainframe at the Manoa Campus at the University of Hawaii, they were typing in card images and hitting return and submitting them. He assumed that they would go on typing no matter what happened to the network. So after sending this card image, if you didn’t hear back you would just keep typing. Well, that’s not how users behave. I decided to do a new model, a theoretical sort of thing, of the operation of the ALOHA Network, fixing these two assumptions. That is: finite model, N users -- for some N, that could be four -- and with blocking. Meaning: if I didn’t get an answer I would stop typing until I got an answer. I did the math and it’s not really… It’s okay, for Ph.D. kind of math. I discovered that of course the ALOHA channel behaves roughly the way Norm said it would; in other words my refinements didn’t really change the answer that much. A little bit, but not much. That is, for large enough N’s and low enough receive… 17% was the throughput of an ALOHA channel, and I got something like 17% too. But not quite. In the course of doing this model I noticed the following interesting thing. I noticed that when it came time... You may remember that the way the ALOHA network worked is that one of these terminals would form a card image and send it in on a radio channel and then wait to see if an acknowledgement came back on the return channel. The return channel was a broadcast channel that went to all the terminals, and the incoming channel was shared. So there was some possibility that two terminals would send in card images at the same time, thereby scrambling each other so that an acknowledgement would not come back on the return channel. Then the terminal had to retransmit its card image. Of course what Norm’s brilliant invention was that the way the terminals would retransmit is that they would wait a random amount of time before retransmitting so that two terminals that happened to collide would not very likely collide again because they would choose different random numbers. When going to do the model -- and I did the model not only in math but I did it in simulation, I was into modeling and I wrote little BCPL programs that simulated this -- there came a moment when I had to choose the random number for the retransmission. Random numbers generally have a mean and a characterization of the random process. What I noticed is in Norm’s model there was no parameter to describe the mean of the retransmission interval. It was not important to his model. It was lost in the tractability assumptions. Huh. That’s interesting. I started playing with the mean of the retransmission interval of this ALOHA channel, and discovered that it mattered. That is, if the retransmission mean was too small then the channel very quickly got unstable with any traffic at all, and if it was very large the throughput of the whole network was very low. So huh, I had discovered this variable, obvious variable, the mean of the retransmission interval. Then I noticed something else, I noticed that…. Anyway, that became a chapter in the thesis which Harvard subsequently accepted.

Shustek: But this network was very unlike the ARPA network that the rest of the thesis was about. How were they related?

Metcalfe: No, no, no. I know how it seems that way now, but at the time packet switching was the theme. The name of my thesis, brilliantly named I might add, is “Packet Communication”. I spent a lot of time naming it, and then I spent much more time on the abstract, and then a little time on the chapters. Because I knew no one was ever going to read the chapters, so the title had to be right. My thesis, by the way, “Packet Communication”, is currently available on amazon.com.

Shustek: Not many theses are.

Metcalfe: No, it’s survived. You can buy it for 30, 40 bucks, hardback. It’s really amazing. The long tail is really long. What was I going to say about this?

Shustek: About adding the ALOHA analysis to the ARPANET design, which seemed to be different.

Metcalfe: There were just chapters about store-and-forward packet switching versus broadcast packet switching, and the different techniques there. Then there was a software level that went on top of all that, all the way up into the operating system. So no, it does sort of hang together as variations of packet switching, although one’s broadcast and one is store-and-forward. Those are just two different modes of packet switching which I was able to unify beautifully in this award-winning dissertation. Phew! Harvard accepted it, and my parents did get to come to Harvard Yard in June of ’73 and watch their son. The first member of our family ever to attend college in the end got his Ph.D. from Harvard.

Shustek: It must have made them very proud.

Metcalfe: Making my parents proud is probably the fundamental motivation of everything I do, subconsciously at least. I did it again recently, and I should get to that point a little later. Where were we?

Shustek: You had [Norm] Abramson’s paper and had done the analysis.

Metcalfe: I got my dissertation accepted. Then Butler introduces me to Charles Simonyi and says, “Bob, we’re building these personal computers. We’d like Charles to write the editor, we’d like you to do the network.”

Shustek: How much of the design had been done at that point?

Metcalfe: He had sketches and designs, Charles did, of how he would build these high speed packet switches. He gave them to me. It wasn’t a detailed design but he was pretty far along. He’d been working on it for some months. I looked at it, and I had just finished doing this ALOHA thing, and I got a bee in my bonnet that there were just “too many notes”; that is, it was just too complicated.

Shustek: Yet it was a different version of something that already worked and that you had worked on, the ARPANET.

Metcalfe: Yes, but it was a lot of mechanism for within a building. And the 50 megabits per second was a bit of a stunt. That’s a really huge number in those days. We were using terminals that ran at 300 bits per second. That was the Texas Instruments Silent 700 that we were all proud of. It's this big, had a acoustic modem that ran at 300 bits per second. So 50 megabits per second -- and I'm not saying 50 gigabits per second, I'm saying 50 megabits -- that was kind of fast. It just looked like there was just too much mechanism there. I had just finished doing this broadcast analysis, and wrote a paper for the Communications -- not the Communications, some branch of the ACM -- on this model I had for a broadcast channel with blocking, a finite population model with blocking.

Shustek: Had you had any collaboration with Abramson in Hawaii?

Metcalfe: Yes. As soon as I read this paper I convinced my management at Xerox that I should spend a month in Hawaii. So Xerox PARC sent me and my wife to Hawaii for a month. I lived on, right on Kala Kowa there in Waikiki and every day drove over to the university and hung out with Norm and Frank Kuo and Charlie Bass and a bunch of other networking guys, and it was a great thing. Then I came back. I'm a little vague about the timing there. But when it came time then to pick up the Signet project I decided… I'm surrounded by Chuck Thacker and Butler Lampson and Charles Simonyi and a bunch of other guys; I really don't want to get into a fight about this. I decided that we should build a bit-serial bus. Instead of "home–running" every terminal to this room -- we had these in the building and they were horrible rats' nests of cables and I took photographs of them -- I decided we would build a network that didn't have this centralized feature. This would be a completely distributed network, maximally distributed. Tthat was the goal: maximally. So we -- I -- started fiddling around with how to do that. I remember I took, I bought, I ordered up some coaxial cable, a thousand feet of it, a spool. Took it down in the basement, hooked up a signal generator, started launching pulses down this cable, just like the pulses in the acoustic delay line memory of that computer we almost built at MIT. Lo and behold, out of the other end of this thousand foot of cable, wiggles came out and they were vaguely like the- not exactly but they were vaguely like the… A big event happened while I was down there that day. This was coax and I needed to put connectors on the end to hook it up to the oscilloscope. I wanted it to be a high quality connection so I could see all the capacitances and the inductances and all that. I wanted to see what it really looked like. I started putting connectors on the ends of this coax. And there was another guy, hippie type guy, and he was wearing a Brooks Brother yellow button down shirt, just like I used to wear. It was David Boggs. He was bringing up Nova 800's as a technician. He was a grad student at Stanford. He saw me trying to put the connectors on the cable and he said, "I know how to do that, let me do it." It involves skinning it and clamping it and crimping it and soldering it and a bunch of things that he had done a lot of and I hadn't done much of. In that moment we became buds. Boggs and I formed our partnership right there. We started launching [bits] and we could see that you could send digital bits down a cable. Not a breakthrough discovery, but for us, we got really practical about it. On May 22nd 1973 I wrote a memo, in which Ethernet was named. Formerly it had been called, we had been as a working name, calling it the Alto ALOHA Network. In that memo I renamed it the "Ether Net" with a capital N and a space: Ether space Net.

Shustek: Explain the derivation.

Metcalfe: Well, this was the derivation. The goal here was to have it, the network, be maximally distributed. These work stations, these Alto, these prototypical personal computers, were going to be one on every desk. Can you imagine that, a computer on every desk? Wow. Very controversial in 1973. Why would you want a computer on every desk when you have this timesharing system down in the basement? Why would you want a computer on your [desk]? What possible use could there be for such a thing? I remember people had that discussion. But I was bought in to maximally distribute it. I wanted as little in the middle of the network as possible and everything on the edges. The middle was going to be the medium, was going to be this cable. We settled on coaxial cable. This is where my friend David Liddle comes in, because David Liddle, of some repute, worked there also. He arrived at PARC a month after me in '72. We became buds. He had in college done some cable television installations, and he was familiar with a thing called the tap, the Jerrold Electronics tap that would puncture the coax and grab hold of it, without cutting it. That's why we chose coax. We said, gee, if we could have this cable running down the middle of the hall and we wanted a PC to hook into it, let's use these passive taps. You drilled a hole, puncture it, connect up to the center conductor in the shield, then you were connected, and you didn't have to break the network to do that. We liked that idea, and that's why we chose coax. Only for that reason, because there was a preexisting device, the "vampire tap" it was called, that would allow you to puncture it without cutting it. We could have used twisted pair, and we could have used optical fibers, or radio, although we decided against radio.

Shustek: Why not radio? That's what Abramson had done.

Metcalfe: Abramson had done radio. It's very simple why we didn't do radio. Trivial, very simple. The ALOHA network ran at 4800 bits per second, or maybe 96 on a good day, and the size of the modem was approximately six feet high and nineteen inches wide, and it cost a lot of money. I don't really remember the numbers, but like that. This network had to fit on a printed circuit card approximately this big [10 inches] because that's how big the Alto card was going to be. So there was no chance of it being a radio. Plus, we were at the same time starting work on the world's first laser printer, which we took a Xerox… We? It was mostly Gary Starkweather and Ron Ryder, but I ended up writing the operating system and the networking for it. This printer that was a laser printer, and it was a page per second, 500 dots per inch. You multiply 500 times 500 times 8½, times 11 per second and you get a big number. So it couldn't run at 4800 bits per second or the printer would be stalled half the time waiting for the bits to come in. It had to run in hundreds of thousands of bits per second to serve the printer. So radio wasn't going to work. So that's why we went to either twisted pair or coax, and settled on coax. But while making that process I decided that Ether Net would work on any medium. Because all we required was that the medium be passive, omnipresent and a medium for the propagation of electromagnetic waves. That's all it needed to be. It could be coax. It could be twisted pair. Instead of calling it "Coax Net", I wanted to think of something more abstract than that. I remembered from freshman physics at MIT about the Michelson and Morley experiment, in which these two scientists performed this very famous experiment in which they proved that the luminiferous ether did not exist. That is, until around 1900 it had been theorized that the way the light reached the earth from the sun was that it was carried at 93 million miles through some medium yet to be detected. There had to be a medium. Air carries sound, there must be this ether between the earth and the sun, and Michelson and Morley through this clever experiment detected that there was no ether There's no such thing as an ether The luminiferous ether did not exist. And what was the luminiferous ether? It was omnipresent, completely passive, and it served as a medium for the propagation of electromagnetic waves. Ta-dah. Ether Net. So I recycled this word and called it Ether Net on May 22nd, 1973, and then laid out its general method of operations, which was a variation of the ALOHA network. But some important variations. From time to time over the last 30-some years various people have said that Ethernet really is ALOHA Network and that Norm Abramson really invented it, and that's fine. Great. The differences between the ALOHA Network and the Ethernet, there's a long list of them and I'm proud of them. But it does have randomized retransmissions and that was the idea that we stole. Well, packet switching we stole from the ARPANET, and randomized retransmissions we stole from Norm Abramson, but then the rest is completely different. So Boggs and I then built the first card, an Ethernet card, to plug into this Alto that Chuck Thacker and Butler Lampson and Ed McWright and others were building. I had the same problem by the way with the Alto that I had with MAXC, which is I'm coming in right behind the guys who got the computer to work, and there are just a few bugs left in the computer. Then a dispute arose as to whether the bugs was in Boggs' and my card or in the Alto, which was still being debugged. Chuck and I again had a run in.

Shustek: Did you discover yet another bug in the computer?

Metcalfe: I don't remember it as vividly as munger. But I do remember I did the following really bad thing, that I'm really ashamed of. So I'm having this dispute. "Chuck, I need your help, your computer doesn't work". "Bob, you got your Ph.D. now, but screw you, you don't know how to debug, you're an idiot". I'm paraphrasing, and that's just my view of it. Obviously Chuck is completely innocent and I am the cause of this problem. But it got so bad. There was a movie popular there in which the tagline of the movie was, "Love is never having to say you're sorry". It was popular then. I made a rubber stamp that said, "Reliability is never having to say you're sorry", and I started stamping my memos with it This drove Chuck Thacker bananas, because I was accusing him of building unreliable computers. I really regret doing that. It was really senseless stupidity on my part, and just part of that obnoxious streak that I have. Anyway, we got it to work.

Shustek: Give me some insight into the management at Xerox PARC at the time. Did you have to get anyone's permission to abandon Charles Simonyi's Signet and take up this new design instead?

Metcalfe: I'm glad you asked. No, I didn't really need permission, but the following thing happened, which involves Chuck Thacker again and answers your question about management. Boggs and I are off developing Ethernet and we needed some help with the analog portion: the inductors and the capacitors of driving the actual cable. Because you had to be down there at picofarads, one or two or three picofarads, and that's a specialty. We had a guy helping us do that and he wasn't up to it, but we didn't know he wasn't up to it. We were having trouble getting it to work. During that period -- this is a period of a month -- Chuck Thacker sees that we're having trouble getting it to work. He writes a memo proposing what he called Xnet, a different net- not signet, not Ethernet, but yet another network called Xnet; 16-bit parallel network with a cable this thick that he felt was the correct solution to networking these Altos together.

Shustek: Home run wired again like the original proposal?

Metcalfe: No, daisy chained.

Shustek: Daisy chained.

Metcalfe: Multiple conductors; the worst possible idea you could have. If you're going to fill a building with computers, you can't have cables like this running around. But Chuck saw his moment. That is, I was not succeeding and he saw a chance to propose this other network, Xnet. But Chuck Thacker is the senior guy on this project and I'm the new guy who just barely got his Ph.D. I'm about to have all the oxygen sucked out of the environment. The technicians will start working on Xnet and everyone'll start thinking Xnet, and the two machines, they'll have this big thick cable and my little project will just die. I did the only thing you could do, is I went and complained to management. I went to Bob Taylor. And by the way, Bob Taylor denies that this ever happened. So there you have it; it could be I dreamt this. I went to Taylor, "Boo-hoo-hoo, Chuck Thacker is about to screw me and he's wrong. You cannot fill a building with cables like this. Trust me, 16 bit parallel is not the way to go. It has to be bit serial. Trust me. I'm a networking guy. Yes, I know we're having a little trouble getting our little wiggles to go fast enough through the cable, but we're going to solve it." Bob Taylor, once again, denies this happened: he made Xnet go away. He convinced Chuck, probably with Butler's help, that you really can't string 16-bit parallel cables all over a building. They're too expensive, they're too unreliable, they're too thick and awkward. This bit serial Ethernet thing is the way to go. "Stop Chuck." And Chuck stopped. Then we switched analog guys to a man named Tat Lam. Tat Lam got it right, and got the puffs [picofarads] where they needed to be, and the rest is history. There were a quarter billion Ethernet ports shipped last year.

Shustek: Who else was involved? You and Boggs were the main drivers of this. Did Thacker have an involvement in the design, did Lampson?

Metcalfe: Yes Lampson and Thacker. When the Ethernet patent got filed, the inventors were listed: Metcalfe, Boggs, Lampson and Thacker, and I think that's completely correct. We did file a patent on it which has long since lapsed.

Shustek: What were their contributions, Lampson and Thacker in particular?

Metcalfe: Every minute of every day. We were all working together on things. We would discuss the packet formats and the voltage, the Manchester encoding. It was a collaborative effort. I remember when 3Com Corporation later went public, 1984, March 4th, we had to write who invented Ethernet. It had to be in the prospectus. There I am in a room full of a dozen lawyers, each getting paid $500.00 an hour, trying to decide whether I was the inventor of Ethernet, the principle inventor of Ethernet, an inventor of Ethernet, or a guy just hanging around in the building the day Ethernet was invented. We considered all the possibilities We settled on principle, and the evidence was that the patent was Metcalfe, Boggs, Lampson and Thacker. It was not alphabetical, they had been "et al"-ed, as they say. Really lame. But no, Chuck and Butler were extremely important in this whole process. I mean, they were the computer designers of the day.

Shustek: In large corporations if you start a project like this, somebody writes specifications that say it has to go so fast and support so many nodes. Did you have any such specification for Ethernet?

Metcalfe: Yes, we didn't.. We were not an engineering organization, so we didn't write -- as I learned later -- we didn't write a marketing requirements document, nor did we write an engineering design document, or any of those important process pieces. But as I mentioned earlier, we did know it had to run hundreds of kilobits per second because the printer demanded that to be kept busy. It couldn't be slow. Here's how we settled on, for example, the speed. The round numbers were it was going to go a mile -- this cable had to go a mile -- it had to run at 2.94 megabits per second, and it had to have up to 256 nodes on it. So we had an 8-bit address, wire wrapped on the backplane of the Alto. That was the 256. We had a mile because that was a round number and we could do that; that is we had the spools and we could send the signals and they came out recoverable at the end of the mile, so that was good. The speed one is the really interesting one. How fast should this network go? It had to be hundreds of kilobits per second. Boggs and I are designing this little card and we start putting the registers on it and the flip-flops and the controls, and the card's filling up. Oh, we need a clock. We need to clock these packets onto the ether- how fast is the clock going to tick? But gosh, golly, gee, we just finished putting the CRC chip on a board extension that Boggs cleverly came up with. A CRC chip, but kind of on the margin of the board where it wasn't supposed to be. Where were we going to put the crystal and the associated electronics for the clock? We said, wait a minute, here's this pin coming off the backplane, and there's a clock for the computer, the system clock, which ticks every 170 nanoseconds; a little 25 nanosecond pulse every 170 nanoseconds. We had decided to use Manchester encoding, which meant two ticks per bit. That turns out to be 340 nanoseconds per bit, which turns out to be 2.94 megabits per second. And that was how we arrived at the speed. Because we didn't have room for the clock; we had to borrow the clock.

Metcalfe: So the way we decided that the Ethernet would run at 2.94 megabits per second is because we used the system clock of the Alto because we didn't have room on the board for a separate clock on our board. And God knows what speed it would've been had we had room. So for years afterwards whenever I was asked "How fast is your Ethernet thingy?", I would say 2.94 megabits per second, and they would say "3 megabits per second", and I would say, "No, 2.94 megabits per second". Hmmm, that's Metcalfe being nerdy or something. I said, "Well listen, how fast are those transcontinental trunks that make up the ARPA computer network -- later the internet -- how fast are they anyway?" "Well, they're 50 kilobits per second." I would like to point out that when you round from 2.94 to 3 megabits per second, the rounding is bigger than the speed of the ARPANET's trunk circuits. So I refuse to round. It's too big a number. 2.94 is the correct way to put that. So years later when, let's see, that would be '76 when the first Ethernet paper got published, July of '76 in the Communications of the ACM, I ended up on a panel at a trade show where the other two panelists were not scientists, they were engineers from companies, and they were busily debating whether you needed 9.8 kilobits per second or whether 1200 bits per second was plenty. I remember 1200 bits per second and these new-fangled modems that ran at 9.8 kilobits per second. The guy peddling the 1200 baud modems -- "baud" was still used then -- argued that at 1200 baud the characters went by on the screen so fast you could barely read them anyway. So the 9800 bits per second was overkill. Then it was my turn to talk, and I'm talking about a network that runs at 2.94 megabits per second. And I felt strange, and it was strange. If 9.8 kilobaud was overkill, what about 2.94 megabaud? I had to point out to them that we had more baud than that; we had two bauds per bit, so we were really running at twice 2.94 megabauds per second.

Shustek: So the design specs were: 2.94 megabits per second, 256 stations and a mile of cable. You knew that if you injected a signal in one end of that mile of cable you could retrieve them out the other end. But here you were putting these vampire taps all along. It does something to the transmission line. Were you--?

Metcalfe: Well that was the problem we were having, was getting that to work. I believe the critical parameter was four puffs [picofarads]. So as long as we could get below four puffs, we could meet those specs. But otherwise the reflections, that is, if the inductance of those taps went above about four puffs, the impedance disturbance introduced by the vampire tap would cause reflections, and those reflections would be seen as collisions, which would abort the transmissions. So we had this critical parameter that when you were sending, the reflections could not be mistaken for collisions.

Shustek: Was that a result of theoretical computation, or did you get 256 taps and put them on a piece of cable?

Metcalfe: There was a lot of measurement, there was some arithmetic. I wouldn't say it was theoretical. It wasn't very theoretical. I think there was some arithmetic involved, yes. You take the distance, you take the amount of energy that would be reflected given a certain impedance mismatch, and then you would do the arithmetic again because it would be attenuated, and you would see if that was above the threshold of the collision detector, and as long as that little formula fell below the threshold you were okay.

Shustek: Talk about the first implementation. It had to fit on this little board that would plug into the Alto. What was the technology? Did you microcode this time?

Metcalfe: Yes. The Alto was a microcoded machine, so a lot of the functionality in the Ethernet controller was. There was Ethernet microcode that ran in the Alto. It was an interrupt system that would trigger microcoded interrupts. The transfers of data from the shift register into memory would be done by microcode running on the processor.

Shustek: A word at a time? The microcode was involved in each word transfer?

Metcalfe: Yes. Every 16 bits there would be a wakeup and we would empty the shift register. Now there was a FIFO between. There was an entering shift register that captured the bits as they came off the line, then there was a transfer into a FIFO, which gave you time to get around to empty the FIFO from the other end, because there were some latencies in the interrupt system.

Shustek: What parts were done in hardware? How about the address recognition, CRC calculations?

Metcalfe: Address recognition was done in microcode, the CRC was done in hardware.

Shustek: Binary exponential back off?

Metcalfe: That was done in microcode but there was some hardware assist. How did that work? I'm not sure where we got the [random number]. We got the random number from somewhere in the machine. There was no random number generator; I think we just got it somewhere. No, in microcode we would shift the mask in that would go against the random number to double it with successive collisions. "In microcode" I think is the general answer to most questions like that. So the FIFO, the shift registers and the collision detection [was in] hardware. The collision detection was pretty simple. With a Manchester encoding, what that means is in every bit, half the time the bit is on and half the time it's off. So while we're transmitting into the cable, during the off periods, that is, where it's supposed to be off now because we're not driving that cable, we'd just look, with a little gate, and if it came on during the off period, that meant somebody else was driving that cable too; hence collision. So there was a little XOR [exclusive-OR] against what was being driven versus what was coming in, and if during the off period something was coming in, "eeekkk", abort the transmission. So that was collision detection. Very simple. It was sort of electrical engineering by computer scientists. And it worked.

Shustek: Did you worry at all about the reliability of having a single cable being shared by all these computers and people inserting taps into the cable and doing it wrong and causing the network to come down, that sort of thing?

Metcalfe: We did. That question still lingers, because as you know, as you well know, ultimately we went back to the centralized wiring scheme and we went from cable sharing to switching. But in those days there was a lot of unreliability in those big cable closets where, all these huge amounts of cable, home run cables, would all come and there were all these connectors. So we just got rid of all the connectors, and in particular the tap- the tap was the problem because if that shorted the cable, then there would be no communication for anybody. So we satisfied ourselves -- and Jerrold Electronics had done this in the cable television industry for some years before -- that these taps worked. And they did work. Actually the early unreliability problem was: the very first machines didn't use the Vampire taps, [they] used BNC connectors. The very first ones. At the end were these two BNC connectors with terminators on them. On more than one occasion people would borrow the terminators, and the network would not work, and we'd go around and there it was: the terminator was gone. The terminators are attractive little solid metal things that people liked, for some reason. The first two machines were called Michelson and Morley, and they were in a set of cubicles. Then it got to be 5 and 10. Ethernet was an option. This is another Thacker thing. "If you want to have Ethernet in your machine you just let us know and we'll build one for you, but it's optional."

Shustek: The customers were all internal to Xerox at this point?

Metcalfe: All scientists in the lab. These machines cost $30,000.00. It was all funny money really. But I don't where the idea came from -- maybe from Chuck, who knows? -- that this Ethernet thingy was an option. But Boggs and I built enough of them for the first few machines, so the first few machines all had Ethernet in them. But it was envisioned that when the whole building was full it would be an option. Can you imagine that? For years afterwards I gave a speech in which the tagline was "networking is not an option". It came from that sad experience, having to [say] "What do you mean it's an option?" One day when we were up to ten machines now in a bunch of cubicles, we were all working away on the early Altos and somebody took the terminator off the end of the cable. All ten people stood up, and we ended up looking at each other. In that moment Ethernet became not an option. What we learned is by knowing that you had a network in every machine, you could do things that if it were an option you couldn't do. And Thacker was one of the guys who pursued this hotly. So, for example, he put in the machine, when you weren't using it, it would fall asleep and start a memory diagnostic, and a little box would appear that would move around on the screen so as not to burn in the phosphors. This is in the early days of screens, so this was a new idea. And then the memory would be exercised. This was some of the earliest semiconductor memories with ECC in them. When errors were detected, packets would be sent to another machine that would collect these diagnostics so preventive maintenance could be done on the machines as their memories went south. You wouldn't have thought of doing that if the network were an option. It was just the normal mode that the machine was on the network so therefore this memory diagnostic feature would work. So Ethernet became not an option.

Shustek: As people learned about Ethernet not everyone was wildly enthusiastic. I found this quote from Bob Bachrach who I think was a physicist at PARC who said, "Technically or conceptually there's nothing new in your proposal, and analysis would show that your system would be a failure." Why was he so critical? What were people's objections?

Metcalfe: My mom used to say if you can't say something nice about someone you should just hold your tongue. But Bachrach was a not nice person. Or is for that matter; he may still be around for all I know. He was sort of a grouchy physicist on another floor in another lab, and he read my memo. And this is a physicist. The phrase you didn't catch in his now famous memo -- this memo's posted all over the world because everyone thinks it's a hoot that he wrote this memo -- he said the problem with Ethernet was that it was not quantum noise limited, which is a very physicist thing to say. Of course it's not quantum noise limited! That wasn't the point. But he resented the fact that we were not using every shred of bandwidth that was capable. I told you how we decided on the clock speed. There were gigabits per second of capacity on that cable, and we were using a pitiful few megabits of it. But he was concerned that it wasn't... And he did a nasty thing, and it was sort of an early lesson in nastiness, which is he wrote a memo. He didn't come see me. He didn't say, "I just read your memo and I have a few questions about it". He didn't do that. He sent a memo to my boss's boss accusing me of writing something that wasn't new and it wasn't quantum [noise limited], completely making an idiot of himself. But it didn't feel that way at the time. It felt kind of nasty. The next thing I know I'm in my boss's boss's office having to explain how this guy's an idiot. Which he is, or was. To put it more kindly, he was from a different universe. I was in a computer science universe, he was in a physics universe. We were solving different problems and he should have minded his own business. Or at least come talk to me before sending a memo to my boss's boss, so maybe I could save him the embarrassment. There are many places where I go where that memo is on the wall, pinned to some cubicle -- "the old Bachrach memo" -- because it's just such a great example. Now Bachrach, I ran into him a decade or two after that unfortunate memo, whereupon he was letting it be known that he'd helped invent Ethernet. How he got there I don't quite know, but it had something to do with this memo being really instrumental in fixing some of the early problems with Ethernet, which is a complete hallucination.

Shustek: Success has a thousand fathers.

Metcalfe: Success has a thousand fathers. And Bacharach regrets that memo, and I feel badly for him.

Shustek: There were other people who objected to Ethernet because of indeterminacy or collisions.

Metcalfe: Well, we're talking about 1973. This is decades of conflict about Ethernet. So there's stories upon stories, and the non-determinism one keeps cropping up even today, another hallucination that people have. Today the industrial automation people, although they generally use Ethernet, they're still a little nervous about it because it is non-deterministic. Because I stupidly developed a stochastic model of the operation of Ethernet, and it is non-zero probability that your packet will never get through. It is also non-zero probability that all the molecules of air in this room will all go into that corner long enough to suffocate you. But the number is so vanishingly small, that model's really not useful. I stupidly did this stochastic model, and those guys have been worried about nuclear power plants melting down because the packets are not getting through. The kludges that they use instead of Ethernet, with token passing and other rigmarole, have non-determinism problems much worse than Ethernet, they just haven't been modeled. Do you know how non-deterministic a token ring is when you lose the token? Say goodbye to your token ring for minutes while it figures out it lost a token and then it reconstitutes the token. That's non-determinism, but it's not modeled as non-determinism. This is a religious, spooky thing.

Shustek: It may perhaps not be as bad as minutes but there is an issue there. There are other--.

Metcalfe: Oh excuse me, I hasten to add, there has not been a collision reported on an Ethernet in ten years because we don't do that anymore. Ethernet has evolved. It's now switched, and where there's vestigial collision detection retransmission mechanisms they're not invoked anymore, so it's moot.

Shustek: Let's explore that a bit. Given the number of technological changes to Ethernet -- it's not shared wire, it's now home run; it's not collision detect, it's now switched -- is it still Ethernet?

Metcalfe: That's a metaphysical question. They still call it Ethernet and that's what matters. Since I coined it and I'm generally regarded as the inventor of Ethernet, every time they call it Ethernet, a little something -- it isn't money -- a little something falls into my coffers, so I don't--.

Shustek: A psychic royalty.

Metcalfe: So Ethernet has been evolving and proliferating for 33 years. The things they call Ethernet these days bear little resemblance to the thing Dave Boggs and I built in 1973. That's a fact. So is it the idea of a LAN [Local Area Network], is that what the word Ethernet means? Yes, pretty much. Is it the packet format, the 802.3 IEEE packet format, which kind of, in variations, persists to this day? Is it the idea of everything being distributed? Well, not so much, now that we have switching. Is it the retransmissions? Not so much. What I come to is this: Ethernet is a business model. What the word Ethernet actually means today is six things, it's the six features. You can tell that I've had this thought before. The Ethernet business model goes like this. Six things. [1] It begins with a de jure standard made by a legitimate standards body, in this case the IEEE 802. [2] The implementations of that standard, painfully arrived at over years, are owned by private companies, not this open source crap. 3Com owns the designs of its chips. [3] Fierce competition among the purveyors of the standard with their various implementations. Fierce, ugly, horrible, wonderful competition. [4] Evolution of the standard based on how things look after it gets shipped, that is in the marketplace. Evolution back in the Standards body that came up with the standard to begin with. [5] Maximization of backward compatibility. (Oh, I've skipped one actually but I'll get to it later.) That is, the subsequent versions make maximal use of existing versions. Preservation of the installed base is very important in this business model. So, for example, my Macintosh has one plug and it figures out if it's 10, 100 or 1000 megabits per second auto-magically, and the switch figures out and it all interoperates. And the thing I skipped which should have come earlier in the list is [6] an ethic in the competitive marketplace, where it is not allowed to be incompatible. That is, the customers, the market -- there was an organic thing -- will not allow the competition to be on the basis of incompatibility. If you're incompatible no one will buy your products. You must be compatible. Those six things are what I call the Ethernet business model. I distinguish it from open source. I distinguish it from Cisco, Intel, Microsoft. I distinguish it from the IBM vertically integrated monopoly model. Ethernet is not the only instance of this model, there are many others, like TCP for example.

Shustek: Presumably you can apply this to operating systems, programming languages, applications, almost any product?

Metcalfe: You could, but it begins with a de jure standard and those six things I just listed.

Shustek: This model, though, doesn't have a name. It is the model that Ethernet used, but clearly anything that uses that model is not called Ethernet.

Metcalfe: Right, no; right, no; I admit.

Shustek: We'll get to the standards-setting process in awhile.

Metcalfe: No, no, no, let me say. When people say Ethernet they're referring to this business model. And here's why the word Ethernet has persisted. Well, Wi-Fi used to be called Wireless Ethernet and then some marketing people changed its name, so that's a counter example to the point I'm about to make. But the word Ethernet has persisted for 33 years, despite the fact that everything's changed. I believe it's because of going back to this business model. I think the name will probably persist a little while longer, although it didn't in the case of Wi-Fi, because what happens is when a new technology gets developed they call it Ethernet. Because they are trying and succeeding and borrowing on the power of the installed base, and that includes the people and the processes and the documentation and the general vibes. When HP tried some years ago -- ten years ago -- to improve the LAN, they tried to call it Fast Ethernet, because they wanted it to borrow all that momentum of the preceding 20 years. I wouldn't let them. I'm not in charge of this, but as a columnist at InfoWorld, I said, that's not Ethernet. You guys at HP are screwing up, because there is a fast Ethernet, it's this thing over here, and that's called something else. So they called it 100VG AnyLAN after awhile, and it died. But the new technology called Fast Ethernet succeeded, and it's very different from the Ethernets that went before. So I'm confident that the word Ethernet has some years ahead of it, because people will keep doing this. They keep doing it. If you do a Google search today on Ethernet, there is no let up. The word Ethernet is just being used left and right, and therefore everyone expects me to be an investor in their company. They want the inventor of Ethernet to be an investor in their Ethernet company, and these are companies that do things I have never heard of.

Shustek: But if you look at the Ethernet model, you can apply that model to things that aren't computer communications.

Metcalfe: Sure.

Shustek: So maybe this business model ought to have a more generic name that people can perhaps associate with you but apply to other domains, like operating systems.

Metcalfe: Right, so instead of calling it the Linux model, you call it the open source model; instead of calling it the Ethernet model, you'd call it what? I don't know.

Shustek: I'll have to work on a name.

Metcalfe: But the meaning of the word Ethernet is congruent to this model. I didn't mean to suggest all versions, all instances of this model should be called the Ethernet. But that's what they generally mean.

Shustek: Let's go back to Xerox. The Ethernet now works and it's 2.94 megabit per second instantiation. What about the higher levels of software and protocols? What did they use and how much were you involved in that?

Metcalfe: I was very involved in it. Way back at MIT in 1970, and many other universities, we did various protocols that were ad hoc; in our case we called it the Interim Network Control program. Then came the NCP which got implemented across the whole ARPANET. Then in the summer of 1973, in a conference room at Stanford University at which Vint Cerf was standing at the front and a bunch of us were sitting around tables -- the very same summer that Ethernet was invented, and by the way the same year that the cell phone was invented, 1973 -- TCP/IP was devised. Now when we were devising it, I had for some time been working on a thing called PUP. Vint and I have an agreement: I invented Ethernet, and he invented TCP/IP. That's our agreement. This PUP thing was an internal Xerox thing, and PUP stood for PARC Universal Packet. Here we are at Xerox building the Ethernet, but we had the ARPANET, and we had a predecessor LAN called the Data General Nova MCA -- Multiprocessor Communications Adaptor -- which was a thick cable like this, and we had telephone modems. PUP, the PARC Universal Packet, was to be, and became, an internet packet, a lowercase internet packet, that could wander among these media. So that packet format was defined, and it was to be carried inside of Ethernet packets. So the notion of encapsulating this PUP packet inside of an Ethernet packet was devised. Then protocols were to be built on top using the PUP format, not the Ethernet format. You went up a level. You didn't use Ethernet packets, you used PUP packets, because the PUP packets could then wander through your internet. They could go anywhere. The first protocol to be done was first done at the Ethernet level, and it's in the Ethernet paper; that is, it's a reliable file transfer protocol called the EFTP -- Experimental or Easy? -- something like that, but it's in the Ethernet paper. It just has all the sequence numbers, and how to start a connection, how to end a connection, how to reliably send and reassemble an arbitrary package of bits. Then that was implemented one level up using PUP packets instead of Ethernet packets. Then Telnet was written. So I wrote, for the Alto -- what was it called? It may have been called Etelnet. But anyway, I had done a Telnet implementation on the PDP10 for MIT. I just did one on the Alto, and it used PUP packets to get into the scanner service routines of the MAXC timesharing systems so that you could log into the MAXC timesharing system from your Alto, converting your Alto back into a pitiful glass teletype. So you would just crank up -- what was it called? Let's say Etelnet -- I'd just say "Etelnet" to the command line on your Alto. Then suddenly Tenex would appear on your screen and then you'd become a Tenex user and then you could use the email on the Tenex and be connected. You could then start Telnet again on MAXC, and go from MAXC out into the ARPANET, so you could piggyback your Telnet. No, in that case you'd be concatenating your Telnets, right? You'd use one Telnet to get to this one, and then the other Telnet to get across the ARPANET. So then we did file transfer. We did Telnet. Then later we -- not me, but other people working at Xerox -- did a LAN email, called Laurel, which is really cool. It ran on the Alto -- it was an email system on the Alto -- that also gatewayed into the ARPANET, but it was LAN based so it was fast and neat.

Shustek: Were these protocols proprietary, were they secretive Xerox? What was the attitude of corporate toward all of this invention?

Metcalfe: That's a very long story. There grew a protocol hierarchy. It didn't have a World Wide Web in it, but then again neither did TCP/IP for a very long time. It was PUP when it was at PARC, but then later at Xerox Systems Development, where I transferred after awhile, it became the XNS system -- the Xerox Network System -- which was to be a product. Then it got released as product by Xerox, and then it started getting released as an open source protocol stack. Xerox started going up, and then they hit the printer format for the laser printers. I'm blanking on that name at the moment. But that's where Adobe came in. So Warnock, Geschke, et al, spun out to do Adobe, and Postscript got developed. But there was an internal Xerox -- it worked -- prototype of Postscript, and I think Xerox balked at the publication of that protocol. Somewhere, something went wrong in there, and it's a very long story. But XNS got picked up and soon dominated the world in networking, because Novell adopted XNS. So suddenly XNS, not by thanks to Xerox or 3Com or anybody else but thanks to Novell, of all people, they took XNS and made it into Netware, which then filled the world with XNS.

Shustek: Where was this in development at the time that summer the colloquium happened at Stanford where TCP was being developed? And what was the relationship?

Metcalfe: I don't know that off the top of my head. It was a close thing, and there was a little bit of awkwardness because the PUP paper hadn't been published and I was not allowed to discuss. So the way that sort of went is, without referring to PUP, I participated in this design and got acknowledged in the original TCP/IP paper. It's a bit of a sore point; though I was not listed as a co-author but I was acknowledged as a contributor. But Vint and I have an agreement. So in those meetings I was kind of sharing my experience. We were building internets and the NCP people were thinking about building internets. So we had some experience, Boggs and I and others; John Shoch. We were sharing that experience sort of in a vague way without giving specs for what PUP looked like.

Shustek: All of you were at this Stanford meeting?

Metcalfe: I was there. I don't remember if Boggs made it or not. Boggs was a grad student at Stanford, as was Dalal, Yogen Dalal, who you may have heard of also.

Shustek: Sure.

Metcalfe: Was around then. So I recruited Yogen when he finished at Stanford; I recruited Crane out of Vint's group at Stanford to be [in] the networking group at Xerox -- after PARC, when we in the Systems Development Division.

Shustek: I'm trying to understand your motivation, because you have the XNS protocols for interneting and now these people are developing a similar but different, and presumably incompatible set of protocols for internetworking. Why is it in your best interest to help them get it right?

Metcalfe: Force of habit. I mean, I'd been attending those seminars all through graduate school and so it was just routine that we would--. Vint will correct me on this but I think it was called INWG, the International Network Working Group. This was either INWG or a prototype of INWG or a branch of INWG that Vint and Kahn and I--. We were working on the advancement of the ARPANET. And the notion… Once again you should interview, have you done Vint Cerf yet? You should interview him about this because he has a much more detailed version. But at one point there wasn't a TCP and an IP. There was just a TCP. Then at some point the IP got broken out. There are arguments about the effect that PUP had on that intellectual evolution and how instrumental it was, and I'm not going there because Vint and I have an agreement. I invented Ethernet, he invented TCP.

Shustek: Did PUP have the notion of a separate IP layer and reliable transfer layer above that?

Metcalfe: Sure. Yes, we had a reliable packet--. Well I mentioned the file [transfer], the EFTP. Yes, they were virtual circuits and--. No, the whole… it's very similar. Neither one of us had the World Wide Web, which I think is really cool.

Shustek: Talk more about corporate Xerox's reaction and ability to let you publish and make this stuff available to the world. Did they wait for patents to be filed? Did you feel constrained by Xerox corporate?

Metcalfe: No, not at all. Xerox PARC was run as a university department. We were a little bit careful with patents. There were patent lawyers around, and there was a little care taken as to the proper order of disclosure. For example the Ethernet paper was reviewed and the patents were filed before that was allowed to be published but there was no… It was going through CACM (Communications of the ACM) review, and Gordon Bell was the reviewer. Dan Siewiorek and Gordon Bell were the guys reviewing my paper, it turns out. That was important later, I might add, in the history of Ethernet, jumping ahead to

1979. But no, Xerox was heaven on earth. It was a great experience for me, and there were a lot of lessons in that and many books have been written about these lessons.

Shustek: So the paper was published in CACM in July of ’76, and I think NCC in June of ’76 there was a big exposure of Ethernet to the world.

Metcalfe: I’m sure I gave a talk there, yes.

Shustek: Do you remember what the reaction was of people to it?

Metcalfe: Oh, well I mentioned earlier this panel session where there was a lot of “What do you need this for?” ”A personal computer on every desk, what’s that about?” Because remember this is- there was no Apple computer, there was no CP/M machines, there was no CROMEMCO, there was no--. That was all way later. Mostly it was ohing and awing and wondering. This is interesting toys in a research lab. In the real world why would you ever want to do that? We had fonts by then, and we were starting to print on our laser printer. It was hilarious when we developed the capability to have fonts. You get to choose your fonts in your memos and print them out on this laser printer. That’s taken for granted now. You have a gazillion--. Well that was new. And it was funny the first few years because with this newfound freedom of course every memo had 14 different fonts in it. They looked like carnivals with all these weird fonts and then people finally calmed down and generally don’t use more than 7 or 8 fonts per document now. But prior to that it was all Courier ball. The Ethernet memo written on May 22nd 1973 was written on a correcting Selectric typewriter with an Orator ball. No, it wasn’t even correcting, excuse me, I was using, what’s that?- Snopake, using an Orator ball. I liked the Orator: sans serif, and sort of clean. So I had some familiarity with fonts prior to the laser printer, but they were all typeballs on a selectric typewriter.

Metcalfe: So here we are: 3Com has been founded in ‘79 and there are no personal computers. But eventually IBM gets around to announcing the IBM PC in August of 1981 and we develop a card and we start plugging it into these things. But we have this established company [Nestar, founded by Harry Saal and Len Shustek] way ahead of us. They have servers, they have software, they’ve got cards there, they’ve got experienced senior managers, they’ve got offices in downtown Palo Alto, they’ve got Carol [Saal], they’ve got, you know, everything. And we’re just this dirtball operation, but we did two things that beat that. This is my model of it. One is, we did not support the Apple II. We told our sales people, if you find somebody who’s got Apple II’s, turn around and leave because they’re going to want us to support the Apple II, and frankly the Ethernet is overkill for Apple II’s and we’re not going to do it ever, so you’re just barking up the wrong trees. Stay away from Apple II’s. Nestar supported Apple II’s. So that sent you guys down this rathole, the Apple II rathole. Then the other decision we made, and I remember arguing with Harry [Saal] about this on the bus a million times, we went into those computer stores and distribution. Harry would always say, “Well, you know, this networking stuff is much too complicated and this has to be sold direct, and we need to be there to help the customer install it.” We said, “Jeez, there’s a lot of computer stores out there”. So we went into distribution and, Bam! Businessland just took us like a rocket into the big time. Businessland, and Computerland, and Xerox-land and IBM-land. So that’s my theory of how we outmaneuvered you guys.

Shustek: As one of the cofounders of Nestar, I can say there's a lot of truth to what you say. We also made some bad bets, like on Token Ring.

Metcalfe: I don't even remember that part.

Shustek: That came later. Let's go back to '76 time frame, the public exposure of Ethernet. We're going to take a diversion. What's happening in the rest of your life? You'd mentioned that you had gone to Hawaii with your wife, so you were married, you had a family. How do you have time to do all of that while you were ...

Metcalfe: I had a wife; I didn't have a family. And I didn't have the same wife that I have today. I've been married to Robin for twenty some years, but I had a different wife then who has since passed away, Amy Blew. She was part of the reason I went to Harvard, because she was a big shot at MIT. She was MIT's draft counselor. So we couldn't leave Boston, so I went to Harvard just for a change of scenery. I might have gone- actually this is funny. I received a letter inviting me to apply for grad school to a university on the west coast. You know, "Hi Bob, we've heard… You should come here. You should consider coming here and we certainly invite your application." Then I looked at the [letter and] there was a logo, and the logo said "Leland Stanford Junior University". And you're not going to believe this, but I didn't want to go to a "junior university". I swear to God, I thought Stanford was a junior university, which meant a two-year thing. I really didn't consider it very seriously. I went to Harvard instead. What a stupid thing. I don't think I'm the only one who has lapses like that, but it's really weird. I didn't realize that the "junior" modified the "Leland Stanford". I think the logo's still the same; I think the word junior is still there.

Shustek: It is.

Metcalfe: But eventually when I took the job at Xerox Park, Amy agreed to go. And she became a big shot at Stanford. She became an assistant provost at Stanford. Then, after these crazy work hours at Xerox and other factors and no children, we divorced in '75, and then she died. She had children, and then she died of brain cancer.

Shustek: Sorry to hear that.

Metcalfe: It really pissed me off too.

Shustek: What happens next? '76, '77, Ethernet begins to be developed and used inside of Xerox. Other networks begin to appear on the scene.

Metcalfe: Well, not really, but vaguely. But in '79, I got to work on the Xerox Star for some years with little or no success, because we hadn't launched the product. I decided to pursue my entrepreneurial ambitions and I left.

Shustek: Before that, didn't you leave Xerox to go to Citibank?

Metcalfe: I left Xerox twice. The first time was in '75, '76, around the time of my divorce from Amy. So I changed marital situations, cities, and jobs, all at the same time -- you know, one of those sort of mid-life crisis early. I went down to work for Citibank in Los Angeles at Transaction Technology and Electronic Funds Transfer, and that lasted about seven months. Then David Liddle, my good buddy, had since transferred; he had moved from CSL, a computer science lab at PARC, across the street to the System Development lab. He could tell I wasn't happy at Citibank and so he got me to go back to Xerox. Then in '79, after four more years, I got itchy again and gave Xerox seven months notice and said I wanted to go out and pursue my entrepreneurial ambitions.

Shustek: Before that, when you went across the street to SDD, did the character of what you were working on change?

Metcalfe: Yes. I became an engineering manager, and our job was to productize all the technologies that we had been playing with at Xerox PARC. I became the manager of System Architecture, and by the way, Chuck Thacker again entered my life because he was my client. I was in charge, in an organizational sense, of managing his microprocessor development effort for the productized version of the product.

Shustek: Even though he was still at CSL?

Metcalfe: He was still at CSL. It was a complicated thing, and so Chuck and I were at each others throats for the third or fourth time there. But eventually the Xerox Star got shipped, it was just after I had left. That's why I recruited [Yogen] Dalal and [Ron] Crane, just to name two, but not [Dave] Boggs. Boggs stayed over in CSL or SSL, one of those L's over there. Yeah, Boggs was not in the Computer Science Laboratory, he was in another lab called the System Science Laboratory, which what the difference is I couldn't tell.

Shustek: Now you have moved from a research lab to a division of a corporation that's building products.

Metcalfe: Yes.

Shustek: Did corporate get more involved? Were there now marketing requirements documents?

Metcalfe: Absolutely. We got serious about product development, with marketing requirements, documents and engineering design reviews, and all that process stuff, but in a Xerox kind of way and not very effective. My analysis is Xerox was a monopoly, and it had never really had to compete, and it didn't know how to compete. We didn't know how to develop competitive products. We just developed cool products, but they weren't in any sense- they were beyond adequate and not very competitive and ...

Shustek: Was the Star a cool product?

Metcalfe: Very cool. You're using it today. It's just shipped by Dell with Microsoft software on it, but yeah, it was very cool. I'm proud to have worked on it. But, you know, "we" -- whenever you talk to a Xeroid, he'll tell you how "they" screwed up. But "we" were there screwing up. I was screwing up in my particular role as manager of architecture. Part of my job was to productize Ethernet. So Ron Crane and I and others started working on a twenty megabit version of Ethernet. Ron Crane can tell you all about how he did that. Then we started productizing PUP as XNS; Yogan Dalal can tell you all about that. But we didn't go very fast very far, so by 1979 we were still pretty far from having products, a year or two after all that effort. It was taking a really long time and I just got itchy to be an entrepreneur. So I worked my way out and wondered off and became a consultant, and then five months later founded 3Com.

Shustek: Did you know when you left that you wanted to do a startup in the networking area, or did you have any strong vision of what you wanted to do?

Metcalfe: No, I just had heard of Steve Jobs, and I'd heard of Bill and Dave [Hewlett and Packard] and I'd heard of Bob Noyce, and it was automatic that in Silicon Valley what you were was an entrepreneur. So I "left to pursue entrepreneurial ambitions". Moved to Boston -- well, I kept my apartment in Palo Alto, got another apartment here in Boston, became a consultant at MIT for Mike Dertouzos in what was then called the Laboratory for Computer Science, but had previously been called Project Mac. It's where I had worked before. It was shortly after I arrived, in February of '79, I was consulting for the Digital Equipment Corporation for a man named Gordon Bell, perhaps well known to you. In the meeting -- and Gordon and I have an agreement about this meeting because we don't exactly remember -- but in this meeting in February '79, out of that meeting came the following idea. Gordon said -- Gordon who had been the referee on my CACM paper in '76, and this February '79 -- says, "We at DEC would like you to design a local area network for us that doesn't run afoul of Xerox's patents". And I said, "I really wouldn't feel comfortable doing that".

Shustek: Even though you were no longer a Xerox employee.

Metcalfe: Right. Just, not my thing. But -- and then this is the agreement we have, we don't know who said the next sentence -- but the next sentence was, "Hey, why don't we just call Xerox and cooperate with them". It was observed that DEC was using Xerox marking engines to make printers, and Xerox was using DEC minicomputers to make their big copiers. Why don't we just [cooperate]. So I drafted a letter to David Liddle's boss, who had been my boss' boss, and David, saying "Gosh golly gee, why don't we work together on a local area network so we can connect your workstations to our minicomputers to your printers and make a network out of this Ethernet thing." Then Gordon signed it and he sent it off, and back came a letter from, I forget his name, but Liddle and his boss, saying "Sure, let's have a meeting." Then in a chance meeting at the U.S. National Bureau of Standards I was also consulting for, I met a guy from Intel who was wondering around NBS looking for applications for a new PMOS process that Intel had. I followed up with him in Santa Clara and met Andy Grove and a guy, Phil Kaufman, who has since died, who was Andy's assistant, and said "Gee, why don't you take your new PMOS chip and make an Ethernet chip with it? And here, I've got DEC and Xerox talking. Why don't you join us and we will make this Ethernet standard". And they said yes! So by June of that year I had arranged, and

Gordon kindly refers to me as the midwife -- or the…he has some term for it -- of this DEC-Intel-Xerox collaboration on Ethernet to make it a standard. That looked like it was going to happen in June of ’79, which is why I founded 3Com on June 4th, 1979 for the purpose of serving the Ethernet compatible market that was sure to develop from the cooperation among DEC, Intel, and Xerox.

Shustek: What did you plan 3Com to do if Intel was making the chips and DEC and Xerox were both building the adapters? What was 3Com’s role?

Metcalfe: Well, I had a long list of products including transceivers, and software, and servers, and switches, and it was much too big a list. It was basically 3Com, Cisco, Nestar, Sun, all in one diagram. I still have the business plan upstairs if you’d like to see it. But it was predicated on the release on September 30th, 1980 of a spec called The Blue Book Spec of Ethernet. I wrote this business plan contingent on the release of actionable specifications in September of 1980 on September 30th, and they came out and they were. Then I started showing this business plan to all the VC’s starting on September 30th and raised 1.1 [million]. I sold a third of 3Com Corporation for 1.1 million dollars in February of 1981, based on this plan to serve what was sure to be a huge Ethernet compatible marketplace now that DEC, Intel, and Xerox had agreed to work together on it. In the fall of ‘79 then, we created project 802, IEEE project 802, for the purpose of blessing the standard that DEC, Intel, and Xerox were developing. Little did we know what would happen next, but Project 802 got created, the blue book got submitted and the meetings started. And with that, war started.

Shustek: Did you get any resistance from Xerox towards releasing this technology that you had developed and was proprietary to them?

Metcalfe: Not really. I’m sure there were moments where there was some discussion, but the calculation was that we needed chips and DEC was the world’s second largest computer company at the time, I guess, or getting there, and we were going to connect our workstations and their minis and our printers. There was a pivotal discussion there. I had recruited Howard Charney to join 3Com. Howard had been a lawyer. He was in fact a patent lawyer. In fact, he incorporated 3Com during this time. Howard had famously been part of a lawsuit where he helped Memorex, as a lawyer, sue IBM for anti-trust violations, a very famous lawsuit of the day. So he was very expert in anti-trust matters. I was getting some resistance from Xerox, which you may recall was a monopoly at the time, about getting together with DEC and having a meeting about this, and Intel too. There was nervousness, and the IBM lawsuit was famous. I called up Howard, I said “Howard, I’ve got have this meeting. How can I have this meeting, legally?” He said, “There’s only five things you can’t do. You know, you can’t set prices, you can’t do da-da-da.” The basic recommendation was convene this meeting with the understanding that the specs will be submitted to a public standards body. Do not have any marketing people in the room, because they will conspire on territories and prices, which is a violation of Sherman and Clayton, et cetera. Oh, have government observers present, and that’s where the National Bureau of Standards, now called NIST, came in. He gave me this list, then I then turned around and gave the same list to the people at Xerox, the people at DEC, and the people at Intel, and they gave them to their lawyers and the meetings occurred thanks to Howard and his very straight-forward advice.

Shustek: Was that the prime motivation for going to the IEEE and wanting to make it a standard?

Metcalfe: Yes. The goal had to be a public standard. We couldn't use it in restraint of trade, so therefore, we, and it's a little murky about how this occurred, there's a lot of disagreement about it. But I'm telling you, we created Project 802 in order to bless this standard we were working on. But then of course, the IEEE is a very careful purveyor of standards–making, so the IEEE processes then took over this thing. That's where all the disagreements and animosities started over all this. Because there were some people who arrived there that weren't so enthusiastic about just blessing standards work that had already been done by somebody else. They wanted to start over. And then in addition to that, two other major companies decided that they weren't going to let this happen, one of them being the International Business Machines Corporation headquartered in Armonk, New York, which is not far from Stamford, Connecticut, which is where Xerox headquarters were. Both of them were office systems companies and hated each other. IBM was going to be damned if they were going to let DEC, the number two computer company, and Xerox, the other office systems company, come up with a standard without their [involvement]. So they walked into Project 802 and said "Hey, wait a minute, we are the experts in computers and we have a network that's much better than this fly-by-night thing called the Ethernet". Of course, they didn't have a network, they just needed to have a network, so they lied. It's later called the IBM Token Ring and it always lagged Ethernet by five years and eventually died, thank goodness. But it didn't die for a very long time, like fifteen years. So they marched in. The other big company that marched in was General Motors, and they had a slightly different angle. "You technologists are designing technology for its own sake. We are a customer, and we are a user, and we have actual applications, and the network you're proposing is not at all what we want. It's for example, nondeterministic, so we're going to design our own network", which became known as the Token Bus. Then there was a big ugly undue-process-of-law type fight, and committees, during which various hilarious things happened. Like, for example, I -- clearly the pre-eminent expert in local area networking in the world because no one else was doing it, I was the only person doing it. They arranged it so that I was not qualified to vote, because I obviously didn't know anything. So, you know, they used IEEE technicalities to disqualify me as a voter.

Shustek: Was it because you weren't part of an established company?

Metcalfe: No, it was even simpler than that. We're a tiny little company; Ron Crane and I are in the company. And these meetings are boondoggles -- they're in London, and they're in Arizona, and they're in Tokyo. We couldn't afford to send ten people to all these meetings, so Ron Crane and I took turns going to the meetings. Then they made a rule [that] you couldn't vote unless you had attended the previous meeting. So, I was disqualified. I just stormed out in a huff, and Ron Crane then continued attending the meetings in our place. But I always thought it was a little strange that the world expert in [local networking] -- because I had been working on it for six years by then and no one else had been working on local Internet working for six years -- and I got disqualified. I thought that was... Anyway, there's a million examples of just nastiness in the IEEE. But eventually the IEEE did the courageous thing, and instead of choosing Ethernet, Token Bus or Token Ring, the IEEE said "We're going to do all three". So instead of Project 802 -- and this is where the dots got invented for 802 -- there became 802.3, Ethernet, 802.4, Token Bus, 802.5, Token Ring. You guys go make separate standards. We're going to have three LAN standards, which we got, three of them, three, four and five. It took decades to kill the other two. Token Bus died pretty early, and then Token Ring hung on forever. That was a very profitable business for IBM.

Shustek: In retrospect do you think it was a mistake to have gone to the IEEE for standards-making? Couldn't you have just had the three companies publish the specification and made a de facto standard.

Metcalfe: No, I think the effort was worth it. This part of the Ethernet model, which is to get an iron clad legitimate de jure standard at the core of it. We ended up with that: adopted by IEEE, adopted by ISO, adopted by… It went right down the main street into Standards Land, and that's a huge asset. But it was a mess. It took two years. By December 1982 Project 802.3 had been created, and then nineteen companies -- without IBM and General Motors hanging around making trouble, led by HP, who emerged as sort of the honest broker leader of this effort -- nineteen companies, Xerox, DEC, Intel, but HP predominantly. Don Loughry was the famous guy from HP in standards-making. In December the nineteen companies issued a press release saying that they were now all agreed on Ethernet. Of course they weren't all agreed on Ethernet but ...

Shustek: This was the end of '81?

Metcalfe: It was December of '82.

Shustek: That's right. December of '82.

Metcalfe: I think. As I remember it. By the way, by then 3Com was shipping Ethernet cards for IBM PC's already, in September of '82.

Shustek: Even though the standard had not yet been finished?

Metcalfe: No**.** And not only that, our cards did not conform to that standard because we had to go to thin coax with onboard transceivers, which was not part of the standard. So, it's a little flakiness there.

<u>Session 2: January 31, 2007</u>

Len Shustek: Len Shustek. It's January 31st, 2007 and we're back, continuing an interview with Bob Metcalfe for the Computer History Museum. Good morning Bob.

Robert Metcalfe: Good Morning.

Shustek: When we left last we were talking about the formation of 3Com. You had met with Gordon Bell in February of 79, incorporated 3Com June 4th, 1979 and the DIX [DEC-Intel-Xerox] Standard was published in September of 1980. Talk about what your conception of the future was for 3Com. What were the products you intended to do, what space were you going to be in and not be in?

Metcalfe: Well, you just finished summarizing February and June and September like "da-da-da-da-da" But those events proceeded at real time, not so quickly. In February with Gordon Bell, at his office out in Maynard, the idea was that DEC and Xerox wanted to connect machines together. Then it was June-sh when Intel joined. The way Intel joined, by the way, is I gave a talk at the National Bureau of Standards on Standardization and Ethernet. There was a man in the audience who came forward and said that he was interested in what I had to say because Intel had a chip technology -- some sort of NMOS or PMOS, probably PMOS -- that they were looking for a use for, and will I come back to Intel in Santa Clara and pitch em on that, which I did. Then a man named Phil Kaufman, who worked for Andy Grove, took an interest, and then Intel joined. It was when Intel joined -- DEC, Intel, Xerox -- that I figured that there was now going to be an industry. I called it in my own thoughts an Ethernet compatible industry. There had been a plug compatible industry with memories and storage. I figured there was gonna be an Ethernet compatible industry. So there would be companyies needed to serve that industry, and that was the general notion. In fact the slightly more general notion was: I had begun to theorize that the rate of adoption of new computer technology was being paced by incompatibility. There's the fact you couldn't connect things together -- SNA and DECNET didn't work together -- and that it was all these incompatibilities that were keeping people from finding computers useful, or making them useful. So 3Com was a slightly generalized notion of the Ethernet; serve the Ethernet compatible market. 3Com was "Computer Communication Compatibility". Its job was to provide compatibility to accelerate the adoption of computer technology.

Shustek: You were thinking at the time not only of the low level hardware connection compatibility, but also of the higher levels software standards, networking protocols, all of that?

Metcalfe: Yes. The three standards that we adopted at inception were: Ethernet of course, but then also TCP/IP, and UNIX. That was 1979.

Shustek: It's interesting that you chose TCP/IP, because Xerox had their own standard for networking protocols which was not TCP/IP.

Metcalfe: Yeah, and later we switched back to the Xerox protocols, and then later we switched back to TCP/IP. But in those days there were battles that ebbed and flowed. It's all over now; it's TCP/IP. But in those days it wasn't clear. SNA -- Systems Network Architecture at IBM -- was the dominant computer communication architecture. I used to like calling it "snah", just to upset them. I remember once going to Raleigh [North Carolina], the center of the universe for IBM's System Network Architecture, and I gave a talk in which I repeatedly referred to it as"snah." At first they were blank faced because they didn't know what I was talking about, and then later they became annoyed when they realized I was talking about SNA. TCP/IP had come out of Vint Cerf's seminars at Stanford at the beginning of summer of 73, which coincidentally was the same summer that Ethernet was invented. By 1979, with Vint Cerf's encouragement, people were trying to make implementations. In thinking about what computer communication compatibility meant, it meant UNIX and TCP/IP; that was the full stack of compatibility. 3Com's first product was the first commercial version of TCP/IP for UNIX.

Shustek: So your first product was a software product, not a hardware product?

Metcalfe: It was, that's right. Actually our first--- I'm glad you put it that way -- our first product was a book. It was called the "3Com Local Computer Network Vendor List". I still have copies. My wife Robyn and I published this book. It was in ‘79; I met Robyn in ’79. So starting in 79 and 80, what we did is I had the notion that there was going be this Ethernet compatible market place for local computer networking. But no one was with the program yet. The term LAN [Local Area Network] had not been coined. I called them “local computer networks”, so the Ethernet paper was a local computer network paper. I decided there was going be this market called local computer networking. So I went out and I took product literature from every product I could find that seemed anything related to local computer networks, xeroxed it in large format to make it difficult to copy -- I don’t know what I was thinking -- and then I wrote a sort of a preface which said “there is this industry called the local computer network industry and here’s proof”. Robyn and I published this book, and it became the first product of 3Com Corporation. It was very useful during venture capital fundraising because the venture capitalists would come through, get my pitch, and leave and never give me any money, but I did tag ‘em for $125 for this book. We had our receptionist set up with a Visa machine, and before I’d let anyone out of the office I’d make them buy a copy of this book. Robyn and I sold 1000 copies of this book for 125 bucks each for the next 5 years. Of course as 3Com started to take off they said we can’t be in the book business, so Robyn and I took that as a personal business. I remember --- speaking of the first product of 3Com corporation -- in the early days, 79, 80, 81, 82, Robyn and I every Saturday would carry an armful of envelopes full of these books with invoices that we had prepared, and we would mail the books out and the money would come rolling in. It was a great little business. The book changed names; it went from the “3Com Local Computer Network Vendor List”, it then became the “Ethernet Handbook”. Five years later we sold it to some big publishing company.

Shustek: I imagine most of the products in the first version of the book were not Ethernet-related.

Metcalfe: Oh no, none of them were. In a way the book backfired, because local computer networks, which then became LANs, started to become hot. Then all those companies that made products that had nothing to do with LANs suddenly decided that their products were LAN products, and I had helped them by including their product literature in my little compendium. So anyone who was selling a multiplexer or a modem or anything, even if it was 20 years old, they would call it a LAN, and they were a LAN company. So in a way the book backfired. But that was the first product of 3Com, which became a personal product for 5 years. Then the real first product was a product called UNET -- “UNIX networking” -- and it was an implementation of TCP/IP for the UNIX operating system. We did this product at the encouragement of Vint Cerf, and the source code came from BBN. That is, BBN had implemented TCP/IP on an ARPA contract and their source code was in the public domain, so we scarfed it up and then productized it, which took a lot of work. I think we invested several hundred thousand dollars in productizing it. We shipped it, and I’m guessing that would be December of 1980, which is also approximately the time that ARPA published the green books establishing TCP/IP as a DOD standard, so there was good timing there. But then there was bad timing, because ARPA had paid a man named Bill Joy to implement TCP/IP for UNIX on the [University of California] Berkeley VAX operating system. So suddenly we’re out trying to sell you UNET, and there’s Bill Joy giving it away as part of the Berkeley distribution. Fortunately the Berkeley distribution only ran on VAX’s, whereas our software ran on VAX’s and PDP-11’s, if you remember what they are. So we had a market, and we sold several million dollars of UNET over the next few years. Meanwhile our hardware products were percolating along. Our second product in March of --- I knew it was March, what year was it? -- March 80, maybe 81, I’m a little vague on this, was an Ethernet transceiver, a 10 megabit per second… It had to be March of 81. The 3C100 was a 10 megabit per second Ethernet transceiver which we fondly called the “brick”. We referred to it as the brick; it was about this big, and it had a transceiver cable D series connector over here, and then a big vampire tap on the other side, and it ran at 10 megabits per second. Ron Crane was the principal developer, and we started selling that.

Shustek: What network interface hardware was it plugging into?

Metcalfe: Good question. 3Com began, in parallel with the development of the transceiver, the development of what ultimately became a family of three Ethernet adapters: one for the QBUS, which was for small PDP-11's, one for the UNIBUS, which was for big PDP-11's and VAX's, and then one for the Multibus, which was an Intel standard. So we had the QE, the UE and the ME were our 3 products. The development for those three products was paid for by Exxon Office Systems under a contract to us in which we granted them the right to own these products, fully paid worldwide perpetual license, non exclusive. It made it a little hard to raise money as the venture capital guys came through that would say "Your technology is owned by the world's largest company? How are you possibly going to compete with them?" And I would say "They're an oil company. They're not going to ever make use of these products. They think they are, but I don't think they are, so you should invest in us. " I managed to convince a few people that that was good, and that's what did happen.

Shustek: Didn't they also have a large investment in Zilog at the time?

Metcalfe: Yes, Zilog was a Exxon company, as I recall. Yeah.

Shustek: When the transceiver came out, were there other non-3Com network interfaces that it could connect to? Who else was making Ethernet interfaces?

Metcalfe: Xerox, the only other maker. In 1980 Xerox announced its ffice system which had Ethernet in it. So in principle our transceivers could be connected, and they were in some cases, to Xerox equipment. But sadly we had a competitor, a man named Tat Lam, who had helped Dave Boggs and I build the first Ethernet to begin with. I wonder what ever happened to Tat Lam? Wonderful man. But anyway, he went into competition -- actually to be fair to Tatt Lam, we went into competition with him, because he was the first maker of Ethernet transceivers. I think he won the Xerox account with his flimsy little lightweight transceiver, and our big heavy transceiver. We didn't win that business.

Shustek: Did you think that making transceivers was key to 3Com's business plan, or were you doing it because Tat Lam's transceivers were not adequate?

Metcalfe: Tat Lam's transceivers were just fine, and so were ours. It was just part of a product family. In the 3Com business plan, which I still have copies of, I had this product road map that was much too big, but it had transceivers and various network controllers and servers and gateways and…. all mapped. The entire computer industry as you know it today was on this piece of paper. Transceivers were on the list and we knew how to make --- Ron Crane in particular knew how to make – transceivers. So we thought we would make them and sell them, and we did. The 3C100 was done by a startup, so it was kind of crude, really. The final manufacturing step is we poured epoxy into it, so it was heavy. We put the epoxy in it for several reasons, which are amusing in retrospect. One was the product was heavy and we were charging a thousand dollars for it, or some huge amount. What was it exactly? $750 or $1500, I forget which; a large amount of money. So we felt it needed to be heavy so people would know they were getting something for their money. Second of all, our first customer was in Japan and we knew that as soon as we shipped it to I think it was NEC, they would open it up and cut it into pieces and try to steal our design, and we thought we'd just make it a little messy for them, slow them down a little. But actually the real reason we put the epoxy, aside from those two made up reasons, was the design when it came time to ship wasn't finished. It had bugs, and in particular there were some transistors -- we called them ballerinas -- they were suspended off the printed circuit card, soldered manually to various points on the card, and it was mechanically unstable. By putting the epoxy in, we locked the ballerinas in place so they wouldn't rattle around. So we shipped these with epoxy. They were a really an interim product which Ron Crane imagined that we might sell a hundred of, or ten of, or five of. We had orders for 20 or 30. Well, we sold 100,000 of them ultimately, in exactly the same form, with the ballerinas in the epoxy. We kept wanting to discontinue the product and substitute another. We did eventually, but it took a long time to get around to replacing because it was a popular product. AT&T -- which you may recall in 1980 was still the monopoly, it hadn't been broken up, '84 hadn't come yet -- we sold these transceivers to Western Electric. After a year of testing and qualifying and all the stuff Western Electric did, they gave it what was called a Kellogg number, a KE number, which made an official product of Western Electric. They wouldn't let us discontinue the product or revise it or rev it because they had qualified that particular product. So we were stuck in a way in the end [with] a difficult-to-manufacture sort of… it worked just fine, I might add. But it required the epoxy to keep it working. Ron Crane's license plate to this day is 3C100.

Shustek: A thousand dollars seems a lot for a transceiver. You then also have to buy the network interface for your particular machine, which cost how much?

Metcalfe: The first Ethernet that I sold cost $5,000 including the controller and the transceiver and the transceiver cable. Remember you had to have a cable from the controller which plugged into the backplane, then you had this cable that went up into the ceiling to where the transceiver was. The whole kit and caboodle was $5,000, which is about a factor of 1000 or more expensive than today.

Shustek: Who else was getting into that business? Who was your competition?

Metcalfe: Well Tat Lam in transceivers. In the general Ethernet business there were four companies. There was Ungerman Bass, Sytek and 3Com, all of which were founded in June or July. In fact I think we were founded a month ahead of Ungerman Bass; we were June and they were July of 79. Sytek, I think, was also June or July. Ralph Ungerman and Mike Pliner and I all talked about forming companies together, but in the end our egos would not permit it, so we each of us started our own networking company. Then there was another company called InterLAN that came along; Paul Severino ran that one. Those are the ones that come to mind.

Shustek: All four of those companies were trying to produce equipment to meet the Ethernet spec? Is that right?

Metcalfe: Well, two of them were. We were all close in competition, but the two that adopted Ethernet most aggressively were InterLAN and 3Com. Ungerman-Bass adopted it too, but they wandered off from time to time into non-Ethernet things. Sytek was not really Ethernet at all; that was a broadband Ethernet offering. They were using cable television type technology. But we were all in the same marketplace rattling around there in Silicon Valley.

Shustek: The Ethernet spec dictated the use of the thick coax, vampire tap and the transceivers. What happened to make it evolve away from that to "thinnet" and the BNC connectors, and how early did that happen?

Metcalfe: So we -- DEC/Intel/Xerox, then later with IEEE -- standardized the half-inch cable, 50 ohm -- I think it was 50 ohm -- and the whole notion of an outboard transceiver with a transceiver cable. Then we started developing the first Ethernet integrated circuit. The story's much longer than this, but we went into a partnership with Seeq Technology, which had initially partnered with Ungerman-Bass but then for some reason that relationship fell apart. We stepped in as the network experts to help Seeq develop an Ethernet chip. That chip came ready, and this is sort of an amazingly fruitful story. In June-ish of '82, as I recall, the chip came ready. We had been looking for what card to put it in, in other words to make the first integrated Ethernet. We had 3 prospects: there was Apple, there was DEC, and there was IBM. IBM said "Drop dead, we're doing the Token Ring, go away". That was the crispest answer. Apple said "Yes, we'd like to buy 300 of them. Would you just now develop them for us, and by the way it has to support the Apple II and the Apple III and the Macintosh." We eventually built that product, but as you could tell it had to do too many things and it was fatally flawed. It was bigger than and more expensive than an Apple II, so it was not a good idea. The really complicated case was DEC, who said "Yes, we'd like you to develop an Ethernet card using your chip for us, but of course we can't buy it unless you meet a lot of our manufacturing and engineering criteria." They tortured us for 2 years and ultimately rejected our product. But in that event IBM told us to drop dead, so our engineering department said "Well you've gotta give the customer what they want. IBM doesn't want anything, forget that. Let's focus on Apple because Apple said yes, and then we'll fiddle around with DEC for a couple of years and it'll teach us how to manufacture stuff". Which they did. This was not satisfying to me, because in August of 81 IBM had introduced it's personal computer. And it seemed, my sense of it was that it was catching on, you know. The Apple II was not going to be the industry standard, that the IBM PC was, actually. I don't know, I had a premonition of some kind that the IBM PC would take off. So what I did is I bought one, and I put it on a table right outside the engineering department's cubicles. I just put it there, and the rest is history. Of course the engineers all came out of their cube--- Ron Crane and -- I keep mentioning Ron but there were others -- they all came out and started looking at this thing, this IBM PC. Before I knew it, it was open and there was a card, an option slot, and it was this big. Our guys are looking at it, and looking at it, and they're, "We could do that, we could." And eventually the company said "well maybe we should do that." This is how the thick coax went away, since you asked. This card had to be this big, and then there was going to be this D-series connector, I think we called it, that went out to the transceiver. We noticed that the electronics in the transceiver -- with this chip from Seeq, we could take the electronics from the transceiver and put them on the same card and still fit in the IBM PC format. Hmmm. That would eliminate the separate box, and the separate PC board, and the driver/receiver electronics to drive the cable that went from the controller to the transceiver, and we could just put it all… But you couldn't take this half-inch thick coax and jam it into the back of an IBM PC, so we had to think of some other way to network it. Ron came up with the idea of using thin coax with BNC connectors, a traditional standard cable television engineering technology. So we put the transceiver on board, and put a little BNC connector on the side of it so that the coax wouldn't be in the ceiling, it would come right down to the back of the IBM PC. This was a little awkward, because we had just finished convincing the IEEE to standardize the half-inch cable, and here we were, high horse standards guys, about to ship something that was non-standard. But we argued coax, coax, same ohmage, and you could in fact hook the thick to thin and it would work. It's approximately the same thing, it's just: look at all these huge advantages. That's when IEEE 802, which had created a committee called "dot 3", 802.3, suddenly began having multiple Ethernet standards. So under the "dot 3" standard there was the thick coax, and now suddenly the thin coax, which we also then made as standard. We shipped in September of 1982 the first Ethernet for the IBM PC, called the Etherlink. We shipped it in September of 82 because that Seeq chip worked. We plugged it into our PC card and it quickly worked. We went right into production with it and launched it in September of 82.

Shustek: So 3Com was really driving the standards process. It wasn't coming from within the IEEE and the committee. You had invented the ThinNet and the IEEE blessed it.

Metcalfe: Yeah. I would say the ThickNet was driven by Xerox, with Intel and DEC sort of going along with the gag, but it was 3Com that introduced the thin coax. Later we tried to introduce twisted pair. Well, we did introduce twisted pair, but a contending twisted pair was chosen as standard instead of ours. So by then we had lost the ability to drive, exactly the way we wanted to, the standard.

Shustek: Now Intel was one of the three companies initially formed to do this standards consortium. Why wasn't Intel producing a chip for the Ethernet that 3Com was using?

Metcalfe: Intel joined the DEC-Intel-Xerox consortium, as I mentioned earlier, in June of 79 and began work on the Ethernet chip. I say "began work" because they didn't, like, begin it and end it really quickly. They began it, and worked on it, worked on it, worked on it, worked on it, worked on it. Their project succumbed to the following temptation. They had this new, I guess it was, PMOS. The reason, by the way, Ethernet ended up running a 10 Mb/s was because the Intel process could run at 10 Mb/s but not 20 Mb/s, which we had proposed internally. So we slowed it down to the 10 Mb/s for Intel, and they began work on their chip. But while they were at it, they said "Well, we can put a lot of transistors on a chip. So why don't we put buffering, and FIFOs, and scatter/gather, and this option, and that option." So the chip became a big magilla, a big project. I think they sent it to Israel to be done and they made it a big effort. And the effort bogged down. Seeq was able to come in and do a chip, a very simple chip that didn't have all the scatter/gather, fancy memory mapping or whatever the heck it was, all the rigmarole that went into the Intel chip. Seeq finished first. Seeq shipped the first Ethernet chip, and we shipped it in the Etherlink as the first Ethernet for IBM PC. Intel eventually came around, and eventually shipped their chip. I mentioned earlier that we had a Multibus card. Our Multibus card and our transceiver were adopted by a new startup. There were a bunch of startups doing work stations. Apollo was one. I remember going to Apollo on Hartwell Avenue [in Lexington MA] in 79 or 80, and trying to convince them they should use Ethernet and TCP/IP and UNIX. But they had started just a little bit too early and they didn't really see the merit of Ethernet or UNIX or TCP/IP, so they started a workstation company that didn't use those three standards. I remember Mr. Nelson had been at Prime and DEC, as I recall, and he was into a Token Ring-ish sort of thing. They developed their own LAN, their own operating system, their own protocol, everything. This other startup got started a little later, 82 as I recall, in Silicon Valley; a company called SUN. Andy Bechtolsheim had come to 3Com, then a graduate student at Stanford, and said he had this workstation that he had been building with Forest Basket over there at Stanford, and maybe since 3Com was learning how to build things, we could build this workstation and sell it, and he'd love for us to do that. We said "No, Andy, we're focused. We do Ethernet cards and software, so if you ever need that, you let us know." He went off and stumbled into Kleiner Perkins and Vinod [Khosla] and so on, and started SUN Microsystems, and then adopted our products. He adopted our Multibus product, and our transceiver product, and we started shipping it to this workstation company. By the way, there were a lot of workstation companies. Apollo and SUN were just two of, I don't know, twenty I'm guessing. Maybe only ten, but a big number, all of whom became our customers for this Multibus card and this transceiver. So that was our early business. Then I began to notice I was on the phone with Vinod Khosla every Monday morning. We'd ship them 50 Ethernet cards the previous week and we were scheduled to ship them 50 this week. But Vinod would call and say "Well, we need 60 this week, not 50", and he would say "Since we're going to buy 60 instead of 50, they should be a little bit cheaper." Then I would argue, "But you didn't forecast 60, you only forecast 50. I'm going to have to expedite production, so they really have to be a little bit more expensive." This price negotiation every Monday. Every Monday. It was a weekly price negotiation with my number one customer. Weekly, Vinod on the phone. He's gone onto bigger and better things. Then I noticed it was 100, and then it was 150, and then it was 200. We would get a thousand dollars, or a fraction of a thousand dollars, on each of those, and he was getting $40,000 on each of those. So, hmm, maybe we should have done SUN after all. That was in the 81, 82 timeframe. We were SUN's supplier and we were shipping. We were not Apollo's supplier, because they had chosen to go their own way.

Shustek: Were you concerned that a manufacturer like SUN might eventually start manufacturing their own Ethernet boards? Did that happen?

Metcalfe: That did happen. SUN told us, "Look eventually Ethernet's so fundamental" -- which we believed also -- "it's going to be on the motherboard. We really appreciate your products and we'd like to keep buying them, but you should know that we're going…" They were very straightforward about it, and we understood. Which led to the following strange decision in 82. We were selling what we call the ME's, the Multibus Ethernet cards, about this big. I forget the exact price, but it was like 5 or 6 or $700 each. That was our main business. We were selling hundreds per month. It was a big deal. With a transceiver and a transceiver cable and some software, often. Our engineers had come up with a way of doing the next generation of the ME. What do you think we called it, the second generation of the ME? We called it the ME2 or the "me too". The decision was should we… Meanwhile we're developing the Etherlink for the IBM PC in the same lab, and we don't have that many engineers. So it was a question of where are we going to invest our energy and priorities? The sales force said "You gotta do the 'me too'". They didn't call it the "me too", they called it the ME2. But Bechtolsheim and company, Vinod and others, are telling us they're going to design us out. Intel, who invented the Multibus standard, they're entering the marketplace. DEC had finally gotten around to introducing Multibus products. And Interlan, maybe not DEC, but Interlan, our arch competitor of the day, had introduced a Multibus controller, and we we're in head to head with InterLan. I remember they were our sharpest competitor. So we decided not to do the "me too" product, and aptly named too. We said "We're going to focus on the IBM PC." It was a great decision in retrospect. We conceded the Multibus market, which SUN was going to integrate into its motherboard. We conceded it to Interlan, and then ultimately to Intel. Of course DEC had gotten around to doing QBus and Unibus adapters, which we were also selling. We conceded that market to them, and we just focused on the Etherlink for the IBM PC and started shipping them in 82. I noticed really quickly that a hundred was a small number. Selling a hundred cards to SUN was nothing like selling a thousand or two thousand or three thousand or five thousand or ten thousand or a hundred thousand or a million cards into the IBM PC market, which is where it went.

Shustek: These were based on the Seeq IC? Or were you also using the Intel IC at the time?

Metcalfe: Well, for a long time we used the Seeq chip. Then other chips came. National Semiconductor had one, and Intel had one; I think those were the principal ones. We taught those companies how to compete with us, in particular National Semiconductor. We taught them how to build Ethernet chips for us, and then they all introduced their own cards to compete with us. Ultimately unsuccessfully, I might add. We warded them off. We fought off Intel, and we fought off National, for a very long time, a decade or two. It was a nasty scene where in order to use their chips, we had to teach them how to design their chips, and then when they started shipping their chips, some genius at National and some genius at Intel said “Well, why don’t we make the cards?” So they started making the cards and we had to fend them off, which we did; as I’m proud to say, we did.

Shustek: Was that the major revenue source for 3Com, network interface cards? Were you also selling software separately? Were there other hardware components?

Metcalfe: Yes 3Com had diversified. The principal revenue generator was the Etherlinks, but we also sold network operating system software, and servers, and gateways, and various software products around it. But the card was the driving revenue product.

Shustek: Where was Microsoft in all of this? When you were writing network operating systems, were they interested in that space? Were they encouraging of your activity?

Metcalfe: Oh yes. Bill Gates and I -- at that time, we were still speaking; I don’t think he likes me much anymore because I have been critical of the Microsoft monopolistic anticompetitive behavior over the years. But in those days Microsoft was 30 people, and we were 15 people. UNET, our first software product, we co-announced with Microsoft. Microsoft --- this was in 79 and 80 -- had decided to make UNIX their strategic operating system. What did they call it? They had a special name for it. Xenix. We, with our TCP/IP software, were going to be their network operating system partner. That is, UNET was going to be part of Xenix. When we announced that, Steve Balmer, who had just joined the company, and I co-announced it. We had a press reception in Woodside, California. Steve was there, and Microsoft, then I’m guessing 35 people, was making a big thrust into the UNIX market with Xenix and we were their networking partner. That was in August of 80. Exactly one year later, in August of 81, IBM announced the IBM PC, and Microsoft’s strategic operating system was no longer UNIX, it was now suddenly DOS. 3Com made the shift at exactly the same moment. We said “Oh my goodness, we thought it was going to be UNIX, but now we think it’s going to be DOS.” We shifted our efforts to DOS. We went from TCP/IP to XNS, and we went from UNIX to DOS. We stuck with Ethernet.

Shustek: Explain why you made the transition from TCP/IP to XNS.

Metcalfe: Xerox at that time was--- there's a lot going on here, but Xerox had begun the notion that it was going to publish its specifications for XNS. XNS was the dominant internet working protocol at that time. There was no TCP/IP. It was in a few universities here and there. It hadn't even become the--- it wasn't even in the internet. The internet was still NCP, Network Control Protocol. So TCP/IP wasn't obviously the choice and Xerox had much more--- Xerox is a bigger company than DARPA. And we knew XNS, and we believed XNS was better than TCP. They're very similar, but we saw certain reasons to make XNS better. Xerox was pushing now for a brief interval of a year or two, was trying to make all of its protocols into an industry standard. We adopted XNS because we knew it -- we had developed it at Xerox -- and because it looked more promising, because Xerox was behind it. So we used XNS to develop our own network operating system to run under DOS.

Metcalfe: In 1979, when I caught wind of the fact that DEC, Intel and Xerox were going to cooperate to make a standard based on Ethernet, and that 3Com should be founded to serve the Ethernet-compatible marketplace in computer communication compatibility in general, hence "3Com". In the valley, there were these other couple of guys who also had vague notions that they wanted to do networking, too. There was Mike Pliner and his cabal getting started. They were sort of out of the National Security Agency, MITRE Corporation. They wanted to do networking. Then there was this guy named Ralph Ungermann, who was from Zilog. He had done some networking with Judy Estrin and others at Zilog, and he was going to start his company. And I was going to start my company. Someone was introducing us to each other, and we were talking about maybe we should start a company together. I remember I went over to Ungermann-Bass, which was -- I guess, it was Ralph's house in Los Altos, in his living room. There was Charlie Bass. There was Ralph Ungermann. There were two other guys, Kennedy and John Davidson, the four of them. We schmoozed, played racquetball. I played racquetball against Ralph Ungermann. I'm a racket sports player, and I've been playing them for many, many years. It didn't surprise me that I could beat Ralph Ungermann, because he hadn't spent his whole life with a racket in his hand, as I had. So I beat him. It's not good to beat Ralph Ungermann at anything. He's a very competitive fellow. That was the end of our idea of starting a company together, I think, because I beat him at racquetball. Maybe that's simplifying, oversimplifying. I opted to found 3Com on June 4th. He had founded Ungermann-Bass. One of the jokes I've loved to tell is I said, "Yeah, Ralph, we should start a company. Let's see. You're Ralph Ungermann, I'm Bob Metcafe. We'll call it Metcalfe and Ungermann." For some reason he ended up founding Ungermann-Bass. Have you noticed that Bass should come before Ungermann, alphabetically? But it was Ungermann-Bass. That could've been the problem. Metcalfe-Ungermann probably didn't sit well with him. Maybe he wanted it to be Ungermann-Metcalfe. I don't know. Anyway, I decided not to call my company Metcalfe anything and gave it a different kind of name.

Shustek: You gave it an unusual name. It starts with a digit. Did that ever cause problems?

Metcalfe: Always, yes. A significant fraction of my entire life to date has been getting the "e" on the end of Metcalfe and getting the "3" on the front of 3Com. It was the number three, capital "C," small "o-m." That was my branding. For years I would have to say that: the number three, capital "C," small "o-m." 3Com: computer communication compatibility. I didn't like the "s" on communications. I would always say, "Computer communication [pause] compatibility," just to emphasize no "s." I don't know why we choose to spend our… Had I to do it over again, the jokes goes, I would've called it Candlestick Networking.

Shustek: Why Candlestick?

Metcalfe: Because years later, after I left 3Com, 3Com paid Candlestick Park, the football stadium, millions of dollars to get it renamed 3Com Park. But if I'd called it Candlestick Networking Systems, we could've saved all that money and just called it… but anyway.

Shustek: So you made the switch from TCP/IP to XNS for the IBM PC implementation. How did that go? Did the product sell well? Were you the dominant supplier in that area?

Metcalfe: It would be an understatement to say the product sold well. We lucked out. The IBM PC caught on, and the channel of distribution appropriate for us also caught on. Much to our good fortune, computer retailing started. There were many of them, Computerland, Xerox had retail, IBM had retail. But the key one for us was a company called Businessland, which was perfect for our product. Businessland set out to serve the high-end business personal computer user, who needed networking. We launched our product through all of them, but especially Businessland. It was that channel of distribution, combined with our product, combined with the success of the IBM PC. Our numbers just took off. We started shipping in September of '82, and we went public in March of '84. The notion of selling hundreds, 50 a week, to Sun and arguing with Vinod [Khosla], that went away. We argued with different people, the retailers. We sold through PC distribution and got a lot of leverage there, and just scaled up. We were selling thousands, and then tens of thousands, and ultimately millions.

Shustek: Who was supporting this product? Networking was new to most of the people buying it. They probably needed some hand-holding. Were you doing that? Or was Businessland doing that?

Metcalfe: Well, we set out initially to do that. We had a support organization. Then we trained a lot of the retailers. Each of these high-end retailers had their own outbound enterprise sales forces and associated support organizations. We trained them how to support networks. That was rough going, because networking was new and our products weren't great. I mean, they were great, but they weren't perfect, so they required support. But somehow it got done. People learned how to use and support the products. And we made them better and better over time. But it took off pretty quickly. It wasn't a long slog. We were profitable when we went public in 1984. I remember, when I first met the Wall Street financial community, they told me things -- taught me things -- that I assumed were rock solid, almost constants of the universe. Like, "In order to go public, you need to be profitable for four quarters in a row with increasing profits and a high degree of confidence in the next four quarters." So we did that, and we went public. Of course since then I've seen a lot of companies go public that didn't have to do any of that stuff. I've always wondered why I took them at their word that this was just the way you had to do it.

Shustek: Do you remember any of the financial parameters at the time you went public, revenue, how much money you raised in the offering?

Metcalfe: We raised $11 million, which isn't very much. In our first venture capital round, in '81, I sold a third of 3Com for $1.1 million. These are round numbers. A year later, we stepped it up two or three times and raised another 2.2 million. Then another year later, we stepped it up some more and raised

4.4 million. Then we went public and raised 11 million, and then a year later -- each of those events was a year -- we did a secondary and we raised 25 million. But the 11 million and the 25, we never touched. Well, maybe we touched them, but that was insurance. Those events, those financing events, were liquiity events more than fundraising events.

Shustek: How big was the company at the time you went public?

Metcalfe: You know, I'm a little vague on that, but I suspect we had 10 or 15 million in revenue when we went public.

Shustek: How many employees?

Metcalfe: I don't remember that. Must've been hundreds, small number of hundreds. I don't remember.

Shustek: What about the upper levels of software? You were selling XNS protocol for the IBM PC. Were you doing servers, file transfer software, all of the rest of it?

Metcalfe: Yes. We offered a network operating system which we called the EtherSeries. We actually did several generations, but the first one was the EtherSeries: EtherDisk, EtherMail. You know, like there was the VISI series: VISICALC, VISI-whatever? We parroted that as the EtherSeries. Our products did PFMTS. "S" turned out to be the important one. There was print, so sharing of printers. Filing, sharing of disks. Terminal, providing terminal access to associated mini-computers, mainframes. Mail, e-mail, LAN e-mail, which was unheard of. And then "S," stubs. I'll explain in a moment. Print, file, mail, terminal and stubs, that's what our software did. That was our network operating system.

Shustek: All this was written within 3Com from the ground up?

Metcalfe: From the ground up, at 3Com, using XNS. "S" was an API that application developers could use. Stub, that's why it was called stubs. That turned out to be the important one. We needed to do our own network operating system, because there wasn't one for DOS. Microsoft hadn't paid any attention to it, because the whole thrust in the early PC movement was the stand-alone, self-contained. You had everything you needed. You had a printer. You had a disk. Everything was all yours. Bill and others just didn't think about networking. Of course, that recurred later. So we had an open field. To put it another way, in order to sell our cards, we had to have software that made use of the cards because no one else was doing that. That situation changed in '85 or so, when a company called Novell came. Novell did a new network operating system, their own: Netware. It used XNS, so XNS had an installed base much larger than TCP/IP for a very long time. It was called Netware. It was called the EtherSeries. 3+Open was another one of our network operating system generations. Novell didn't sell cards. They would use anybody's cards, including our competitors'. The principal technical advantage they had over us, which caused them to win -- they won the software battle; the network operating system battle of the mid-'80s, Netware won. They won it because they did file sharing and we did disk sharing. I should hasten to add that in 1982 IBM came out with their version of their PC that had a hard disk in it, called the IBM PC/XT.

it had a huge hard disk in it, which was quite innovative; it was a ten megabyte hard disk. Our customers wanted to buy very few of those, because they were expensive. No one really could think of how to use ten megabytes of hard disk. EtherLink could be used to share a hard disk. PFMTS: we did disk sharing. You could buy a work group of PCs with one XT, and hook the printer to the XT. Then all these PCs here would use the XT to share its printer and disks. That was the original value proposition, sharing that expensive ten megabyte, did you hear me, ten megabyte, disk. I know we don't say the m-word anymore but a ten megabyte disk, and then a printer. A dot matrix printer, or later, laser printers, would be shared using this networking software.

Shustek: The disk was shared in the sense of partitioning the disk so that each computer remotely had access to a section of the disk. But couldn't see, or could see, partitions of others?

Metcalfe: Couldn't see. Well, initially couldn't, later could. That was the flaw. I wish I had attended the meeting, and I wish I'd had the foresight, where we decided to do disk sharing instead of file sharing, because that's how Novell beat us. They came in with file sharing, and their first application were multi-user accounting packages, which needed file sharing. Ultimately we sort of cobbled that together with our disk sharing software. But they, by that time, had achieved critical escape velocity and they blew past us. It was this funny period where 3Com was easily five times bigger than Novell, but respectable newspapers and analysts would report that Novell had 70 percent market share. Well, that's not possible. But they had software. They had sort of transformed the market to be really not the market for networking but the market for network operating systems, in which they did have 70 percent share. But we were selling all the doodads that went under their software. So their software became very important to us, because it grew the market for our products evermore. We still made the best cards. Ultimately, Novell began making its own cards. We drove them out of that business, because they didn't know what they were doing. Our products were cheaper and faster and better than theirs. Ultimately they went back to the high margin software business. They were helpful to us. It was co-opetition. Ray Noorda, may he rest in peace, passed away recently. Ray Noorda introduced to us the term co-opetition, which meant, "We're competing but we're also cooperating. Let's make that work." And we did with Novell.

Shustek: Eventually you became the hardware supplier, Novell became the software supplier, and Microsoft became the underlying operating system supplier. Is that a fair representation?

Metcalfe: Well, much later, closer to 1990, Microsoft… You know how Microsoft takes the leader in a segment and aims at it and then kills it. Like Lotus, for example. Like Digital Research, and then Lotus. They set their sights on Novell. Novell was selling the leading network operating system, and Microsoft decided to kill them. So they came to us. They said, "Since you know how to do networking and we don't, why don't we partner, and we'll kill Novell together." By kill I mean, in a very positive competitive sense: offer better products that customers will prefer. That's what I meant when I said kill. So we entered a codevelopment with IBM -- IBM, Microsoft and 3Com -- to do the IBM LAN Manager for OS/2. In the course of that, Microsoft stole our products and our customers, and finally drove us out of the network operating system business in a very bitter, horrible partnership typical of Microsoft partnerships. This is when my relationship with Bill became strained. That was around the time I was leaving 3Com. The company wrote off $90 million one quarter because of this defunct partnership with Microsoft, in which they failed, by the way, with our technology, even as good as it was. They failed to unseat Novell in 1990.

Shustek: How much of that has to do with the failure of OS/2?

Metcalfe: A lot of it had to do with the failure of OS/2, but a lot of it had to do with predatory behavior by Microsoft. I remember the guy that we were working with -- wasn't me exactly, but guys in my company and guys in Bill's company did all this together. As the relationship's falling apart… He's a famous guy, I won't mention his name, but he left Microsoft and became very famous. Now he feels about Microsoft the way I do, now that he's not there anymore. I remember once he said, "You made a mistake. You trusted us." Oy. That's what we had done. We had built a contract which they then exploited, and abused us and took our network. But they failed; in the course of damaging us, they failed to unseat Novell for a very long time.

Shustek: Back when Microsoft was telling you that you knew networking and they didn't, did they ever try to buy you?

Metcalfe: No.

Shustek: You were almost bought by Convergent, but that was later.

Metcalfe: That's not accurate, what you just said. We failed to buy Convergent. We were buying Convergent, and that acquisition fell apart famously. But that was later. When was that? Not so later.

Shustek: '86.

Metcalfe: Was it '86, really? We, 3Com, began doing mergers, and we did it really badly. Convergent's the most famous case. We did manage to acquire Bridge Communications, and that was a horrible mess. Then we failed to buy Convergent Technologies, and then we failed to buy Echelon Networks, and ten others that we either succeeded or failed in buying. But none of them worked out very well. It wasn't until I was gone -- maybe I was the problem -- that the company became much better at M&A [mergers and acquisitions] for a long streak, under Eric Benhamou. Then Eric made his mistake, and he bought U.S. Robotics. Merged with U.S. Robotics, and that was a horrible debacle.

Shustek: Let's go back to the earlier days of the company. Talk about what it was like inside the company, what the culture was like. Did you have a strong notion that you wanted to establish a particular kind of company? And how did that change over the years as it grew?

Metcalfe: Well, the culture, the whole topic of culture, was introduced with the arrival of Bill Krause, whom we recruited from HP in 1981 to be our president. He came from HP, where he had been for 14 years. We transplanted the HP culture into 3Com. A lot of the people came from HP into 3Com. So Bill [Hewlett] and Dave [Packard] were the culture mentors of the company. They never came over, but we were management by walking around. We had cubicles. We were just HP through and through. Before Bill's arrival, because of my inexperience in business, all of our business cards had no titles on them, because I had this notion that would be…. We were so small everybody had to do everything. A title was kind of a bad thing. Bill arrived, and immediately we all had titles on our cards. It took me a year to notice that it worked. I had originally thought that title was a form of compensation. You compensated someone by making them important, by giving them some inflated title. That's not what titles are for, I learned from Bill Krause. Titles are to communicate what people do, so that you can relate to them better. So he introduced all these titles, product marketing manager, director of product marketing, channel marketing, engineering manager, director of engineering, vice president of engineering -- all titles that had come right from HP's hierarchy. What I saw is that they worked. They gave the company cohesion and communication. So everyone had titles. We were careful not to use titles as a compensation scheme but as a communication tool. So we had the HP culture.

Shustek: What was your title, and how was your relationship with Bill?

Metcalfe: Well, let's see. Initially, I was chairman, president, chief executive, vice president of engineering, vice president of sales, vice president of marketing. It was just me for six months, so I had all of them. Then I became chairman, chief executive and president. I remember Howard [Charney] was VP engineering and manufacturing. Ken Morris was VP sales and marketing, briefly; seven months. Then Bill came. He became president, and I was chairman and chief executive, but everyone else, chief financial officer and everybody else, reported to Bill. That lasted for a year. Then I became chairman and vice president of sales and marketing. Bill became president, chief executive. That persisted for another year or two. Then I held a number of positions: general manager, vice president of marketing. I started the software division. I started the workstation division. Then I ran all of hardware for a while. I was there at the corporation per se, formally for 11 years. But then there was the year before and the year after, so I was really there for 13 years.

Shustek: Your relationship with Bill sounds like a good working relationship.

Metcalfe: Fantastic. There were some awkward moments. In the transition where he became chief executive, there was a bad month or two. But to both of our credits… You know, it was traditional in those days for the founders of Silicon Valley companies to flounce out after the adult supervision arrived, and I was close to flouncing. Steve Jobs had flounced out and sold all of his stock in Apple. And Steve is one of my gods. It would've been very easy for me to flounce out at that time, but I didn't. I, you know, sucked it up, and for a very good reason. Because in the early founding days of the company, I interviewed venture capitalists prior to raising any venture capital. I took them to lunch. I didn't ask them for money, I just asked them for advice. If you want money, you ask for advice. If you want advice, you ask for money. So I was asking for advice. I learned these three lessons, which I remember to this very day, the three reasons why start-ups fail. One is the uncontrollable ego of the founder. Two is lack of focus, and three is undercapitalization. So then later, when I went back to the venture capital community to raise money, I would say, before they had a chance to blurt it out, I would say, "Here are the three mistakes I am not going to make. A, I have decided that it's more important that this company succeed than that I run it. Two is, even though I have this business plan that shows a million products, trust me, we're going to focus on a few of them. And three, I'm here raising money, because we're not going to be undercapitalized." That worked, eventually. That was another difficult period, but we did succeed in raising first-class venture capital using that sales technique. That is, basically raising the three objections in advance and answering them before they were… Now, what question of yours was I just answering?

Shustek: Relationship with Bill.

Metcalfe: Right. I decided the success of the company was more important than running it, so I set about building a board of directors, a first-class board of directors. I tricked the VCs into this in a way, in a shrewd maneuver. I was out raising venture capital. I'd heard that one of the problems with venture capitalists is that, at each firm there were the experienced guys who had founded the firm, and then there were all these MBAs that worked for them, the associates. If you weren't careful, you'd take money from a VC and then they'd put one of these MBAs on your board. It got important to me that that not happen to me, because I've always resented MBAs. They always got paid more than I did, and I was smarter than them, and I never understood that. I wanted to be sure that I didn't get stuck with any of these young MBAs on my board of directors. When it came time to negotiate the venture capital deal, they always wanted to have, the VCs wanted to have, the right to appoint a director. I just refused. I said, "No. Trust me. I'm going to recruit a good board of directors, and, no, you don't have a right. I will not grant you that right." They didn't like this. My stubbornness was part of the difficulty on this part. But I did in each case. So at Melcor Venture Management, run by Jack Melcor, a fabulously successful venture capitalist who very few people have heard of, I asked Jack, I said, "By the way, Jack, if I asked you to be on my board, would you be on my board?" Now this is right after I had an argument with him about not giving him the right to appoint a director. So he said, "Yeah, sure, I'd do that." Then, at NEA, Dick Kramlich, another famous venture capitalist with whom I've fought over the issue of board composition: "Dick, if I asked you to be on my board of directors, would you say yes?" Dick said, "Sure, I'd be happy to be on your board." Then at Mayfield Fund, where I again fought bitterly over board composition, Wally Davis, who is the guy interested in us. I said, "Wally, if I asked you to be on my board, would you be on my board?" He said, "Sure, I'd be happy to be on your board." Right after the deal closed, I asked all three of these guys to be on my board. So I got three top-drawer VCs and no young MBAs on my board. This worked out beautifully, because these three guys had started 100 companies, and they knew how to start companies. Together, we recruited Bill Krause, which gets around to your question, again. Bill had been at HP for 14 years, was running the computer systems division there that sold MPE, HP3000s. He knew sales and marketing. He had scale. He had 500 people working for him. We convinced him to join the company, gave him a big hunk of equity. Here was my naïve notion. I was going to be the chairman, chief executive and the strategist and the visionary, and Bill was going to be the president, chief operating officer and make the trains run on time. This is a standard, broken model of governance. Very quickly it became clear that Bill Krause was a good choice, and he wasn't going to put up with me being chief executive. I made a mistake. I predicted that the company would go like this [indicates a steeply rising curve] over a certain time period. I was wrong by six months. Only six months. I wasn't that far off. But that six months cost me the CEOship. That, and other events which we could go into, in which Ethernet was doomed. Famous headline: "Ethernet ist ein super flop," out of a German newspaper. Immediately, my revenue went to zero. Anyway, during the zero revenue period, with expenses going up as we ramped our staff with our newfound venture capital, it became clear we were flying toward a wall and the company really needed adult supervision. That's sort of when Bill became chief executive. The company was so desperate for sales and marketing, I became the vice president of sales and marketing. Now, that's really strange. That was a measure of our desperation, which was, "Okay. We don't have any sales. And, Bob, you can't be chief executive anymore, because you were wrong; it didn't go like this [steeply rising]. You're not allowed to be wrong. Bill, he's just better at running things, so he should be chief executive." All of which, in retrospect, was exactly the right thing to do. I turn out to be not a bad salesman, and my board realized that. They also realized that Ethernet was an evangelism situation in 19 -- when did this happen? -- 82; was an evangelism play, and I'm an evangelist. I became head of sales and marketing. Plus I had Bill, who's an excellent sales and marketing guy, as my boss now.

Although I was his boss, because I was chairman. It was complicated. Actually, it was the fact that Bill and the board wanted me to be vice president of sales and marketing is why I didn't flounce out. Because they were saying, "We're just going to change things to give people their jobs that they're best suited to do, and you should really be head of sales and marketing. Bill should really run everything." That made sense to me. Plus, I had recruited this board. I had carefully recruited this board, of which I was very proud, to make this exact decision. How could I flounce out? I couldn't. So I stayed, and it worked out.

Shustek: On the other hand, your background was on the technology side. Why didn't you become VP of technology, engineering?

Metcalfe: Because my aptitude, because I'm not actually a deep, deep, deep, deep, deep engineer, although I've done it. I'm not really an engineering manager. Larry Birnbaum, by then, was our VP of engineering. My specialty in life has always been that, among the engineers, I was the jock. And among the jocks, I was the engineer. And among the sales and marketing people, I was the engineering guy. And among the engineering guy, I was the sales and marketing. I'm always better at the thing you're not good at. We needed me to be the head of sales and marketing. And it worked, that thought that, "Okay, we're going to just send Bob out to get orders and grow a sales force." It was a lot of personal selling. There was no sales force. That was the first thing I did was appoint three salesmen, from inside the company, who had been doing other things. Mike Halaburka had come from HP, and he was putatively our head of sales. But he had none. When I became head of sales, I said, "Mike, you take care of the western region." Then we have one salesman, and me. There was Dave DePew, who we had just recruited as a production engineer from Berkeley. The company was small, so everybody knew everybody, and I knew Dave DePew's father had been a lifelong successful salesman. And DePew was single, so he could move on a dime. And we didn't have any production problems, because we had no sales, so having a production engineer was a bit of overkill. So I convinced him to take the eastern region of the United States. That next week he moved to Washington, D.C. to become head of the eastern region, a guy who had never sold anything before. Actually, I tried to convince this woman in product marketing, her name then was Marlene Martin. Very good contributor, but we didn't need any marketing. We needed sales. I tried to convince her to be central region, and she quit. Dave Colson was another product marketing guy. I gave him the central region. DePew had eastern. Halaburka had western. And Colson, another product marketing guy, not a sales guy, gave him the central region. David Colson was English, and he had an English accent, so I gave him "rest of world". DePew had the central region. And everything outside the United States, that was Colson. DePew had the east coast, and Halaburka had the west coast. Guess what? We started getting orders, by just having a sales force. It was a big breakthrough concept: send people to go ask for orders, and guess what? You get them. They started happening. Then I grew the sales force. The next wave was to get a search firm and then to recruit actual, real, regional managers for three or four regions. That was the next step. Then sales became too technical. Not technical [as in] engineering technical. But sales compensation, territory management, sales tools, dealing with distribution; it became outside of my skill set. Then we went and recruited an actual VP of sales. Chuck, I think his name was. What we learned there is that people have operating ranges. I went from zero to a million a month. Chuck – no, it was Mike -- went from a million a month to five million a month. Chuck went from five million a month to 10 or 15 million a month. Then Bob Finnoccio came along, and he took us to billions. Each of these sales guys succeeded in his range and then had to be replaced as we scaled up. In that story, since I mentioned DePew, DePew succeeded. That was a bad decision with a good outcome. Sent the wrong guy to the east coast, and he succeeded wildly. So much so that his next promotion was not to be the head of eastern region. His promotion was to be the head of the northeastern region. Then he succeeded at that so wildly his next promotion was to be head of the Washington region. Then his promotion after that was to be head of OEM sales in the Washington region. Each of those was a promotion for Dave. His compensation went up, and his responsibilities were going up, if you count from the bottom. If you count from the top, he was being demoted each time. There was an interesting lesson there about how, in big companies, the people grow faster than the company. But in small companies, the company grows faster than the people. The company was going like this [steep curve], and Depew was going like this [shallower curve]. He was getting demoted from the top and promoted from the bottom.

Shustek: How did he feel about that?

Metcalfe: I think he had his awkward moments, where he didn't understand why he could no longer be head of the bigger entity. But then he noticed his compensation was going up, and he was being very successful. We should ask him, but I think he's pretty happy about it. He made a lot of money.

Shustek: How did your role change as somebody else came in to run sales and marketing?

Metcalfe: Well, I had a series of jobs. I went from head of sales and marketing, then briefly I was head of strategy and projects, which didn't last long because it wasn't a real job. Then I was given the job of starting the software division, and then the workstation division. I succeeded at those. The biggest job in the company was general manager of hardware, including adaptors. That was my biggest job. I was responsible for 70 percent of revenue and 110 percent of profits. Then I became candidate for CEO again. This is 1990, and Bill Krause was getting ready to be non-executive chairman. Microsoft had [slams fist to hand] damaged us. We were going sideways. Bill couldn't be CEO anymore. I was running the biggest division of the company. So there were three candidates for CEO again. I was one of them, and the board chose Eric Benhamou to be CEO.

Shustek: Who had come from Bridge.

Metcalfe: Who had come from Bridge. He was the principal asset that we got from Bridge. And then I retired. The board chose someone else to be CEO twice, separated by ten years. Both cases it wasn't me. The second time, I decided to retire amicably.

Shustek: Did you seriously want the job at the time? Did you want to go back to being CEO?

Metcalfe: I aspired to be CEO of my own company again. I think, the board, in both cases, when it chose Bill Krause in 1982, when it chose Eric Benhamou in 1990, certainly in retrospect, both cases were the right decision, because the company blossomed after those two decisions. So I'm not complaining.

Shustek: Were you bitter at the time they chose Eric?

Metcalfe: No.

Shustek: But you left.

Metcalfe: Yeah, not bitterly. I could see why they chose him. He was ten years younger, had been succeeding in the software division. He was a good choice. Bitter would not be the word, no. Disappointed. Then I decided it would be better for me and Eric if I weren't around anymore. So then I began, you know, an amicable retirement. I remember they had a big retirement party for me at the new building in 1990. I went off to become retired and then a journalist.

Shustek: Talk about the acquisition of Bridge that got you Eric Benhamou, but a whole series of other products as well.

Metcalfe: Well, the acquisition of Bridge was extremely unpleasant, and the source of the unpleasantness came from the word "acquisition". If only we had acquired Bridge, things would've been a lot better.

Shustek: Why was it not an acquisition? What was it?

Metcalfe: We decided, Judy Estrin, Bill Carrico, Bill Krause and I, that we would merge our two companies. Now, 3Com is about two or three times bigger than Bridge at that time. But we decided we were going to merge. We liked the idea because we thought we were going to get Bill Carrico to be our chief executive out of it. That was sort of Bill Krause's first attempt to not be CEO of the company. So Bill Carrico became president, and Judy became something like CTO. In other words, we tried to merge the companies rather than acquire them. That was a big problem. Because suddenly we had two heads of engineering, two heads of Germany, two heads of England, two heads of the U.S., two heads of the northeast, two heads of the southeast. What happened for the next two years is those pairs of people started trying to kill each other. So it was extremely unpleasant. During that two years is when Cisco took off. Had we done that better, there would be no Cisco, and the world would be a better place. [Laughs] Just kidding; that was just a joke. Everybody at 3Com has now gone to Cisco, so I have to be careful.

Metcalfe: I was living on what is now called Sand Hill Road in Palo Alto, had an apartment there; had lived there for a long time, five years or so. I decided that I was going to start a company. I started meeting with all the venture capitalists on Sand Hill Road, just asking them how to start a company, and I got a lot of advice. Reed Dennis was the first VC I ever met with, but then I met with all of the others: Bukowski [ph?], Kramlich, and Tommy Davis -- all the big names -- who were all kind enough to have lunch with me, or breakfast or dinner; I gained a lot of weight during that period. Then it came time to move out of my dining room and we found some sublet space cheap at 3000 Sand Hill Road, which was ground zero of venture capital, and then decided that we would write a business plan to be ready on September 30, 1980 when DEC, Intel and Xerox were about to publish the spec for the Ethernet bluebook specification that was later to be submitted to the IEEE Project 802 for validation, and so on. That story turned ugly, too. So I had my business plan on September 30th and I went back to all these

VCs and got some term sheets, but my partners and I decided that we wanted top dollar for our stock. I remember we got a term sheet from Mayfield, and Venrock and NEA, as I recall. I think that was it. Let's say it was for $13 a share. I think I remember that number. We decided that we could do better than that. We'd much rather go back to our jobs at Xerox than put up with that kind of money. That would be selling a third of our company for $1.1 million, and we thought that was too low. Especially since we already had products that we were just beginning to sell. So I went off… I don't want to name names here, but another VC firm promised us $21 a share so we went with them. Then over a long period of time -- months, it was a month or two -- they failed to close because they couldn't find anyone else to join their syndicate. Funny, everybody but them thought we were worth $13 a share, and they thought we were worth $21. They started weaseling, and they weaseled for too long. Let me not spend, uhhg, it's a whole other story about how they weaseled, and now that I'm a venture capitalist I can see why they weaseled, but they weaseled. So I called up Jack Melchor, who had been… No, he called; his office called me, and said -- unbeknownst to this whole other process -- "We've been doing our diligence on your company now for six months, and we'd like to talk to you about making an investment." And here I am with this weaseling VC. So I went over to Jack's office and I sat in his office and he said, "We'd like to invest at $13 a share," at the old number. I said, "Well, okay, let's do that." Because Jack represented all the major CEOs in Silicon Valley, Bob Noyce, and Ken Oshman and John Young, and it went on. He was a very classy guy. So he was <inaudible> and he wanted to be the lead investor so he was going to invest $400,000, but he wanted a couple of other firms to join him and he asked me if I had any suggestions. So I said, "Well, one of my constraints is that this weaseling VC not be included." He said, "Oh, that's odd. Okay. They're a pretty good firm." " Yeah, but my relationship with them is broken. I do not want them associated with my company." I said, "Well, there's Dick Kramlich at NEA who I like, and then there's Wally Davis at Mayfield who I like, and then there's Venrock. Venrock wants to invest, too." And Jack said, "Well, you have to choose two, because I think this should be a three-way syndicate." So I chose NEA and Mayfield, and not Venrock. Venrock came in on the second round. But here I am saying no to Venrock for the second time, which is not something you do. Jack picked up the phone as I sat there. "Dick," Dick Kramlich, "this is Jack Melchor. Bob Metcalfe says you're interested in the company. Would you like to join my syndicate?" Dick said, "Yes." "At $13 a share?" "Yes, I've already made that offer," Dick says. "I'm going to lead at 400," says Jack, and "How are you good for 300 or 250?" And Dick says, "Yes." "Okay, thanks, Dick." Picks up the phone, calls Wally Davis. "Metcalfe says you might be interested in investing at $13 a share. Are you in for 300?", whatever the number was, "$300,000?" Wally Davis says, "Yes." Click. Deal's done. Then a month later, the money rolled in, a million one, like this [holds an imaginary check], for a million one. That's how I raised the money.

Shustek: A million one for 30 percent of the company?

Metcalfe: Thirty-three percent of the company, a million one. We started the fundraising on September 30th and the money hit the bank on February 21st of 1981. Then I got Melchor, and Kramlik and Davis, three of the top VCs, to be on my board of directors. A year later we raised a second round with a substantial step up, two or three times. We raised $2.2 million and that's when Venrock and Sequoia came in, two very classy firms.

Shustek: Did the VCs stay on your board for a long time?

Metcalfe: Yes. I remember Pierre Lamond was an observer. He later was significant in the founding of Cisco. Wally Davis left Mayfield, so he was replaced by Gib Meyers, who stayed on for quite a long time, and Dick Kramlich. After we went public, they were still on the board.

Shustek: Let's talk about Cisco and why it is they developed, and why 3Com didn't become what Cisco has become.

Metcalfe: Well, there's a lot companies out there that shouldn't exist if I knew what I was doing, and Cisco is a prominent example. 3Com was in networking way ahead of Cisco, way ahead of Sandy and her husband at Stanford, Sandy Lerner and…

Shustek: Bosack. Len Bosack.

Metcalfe: Len Bosack. We were in networking way ahead of them, so they really shouldn't ever have existed as a company. Then later we bought Bridge Communications, which was an actual routing companies. So we, 3Com, should have been Cisco, actually. But except for me; there was some blindness that I had, I guess. One of my blindnesses, I remember, was I was running a lot of marketing at 3Com and there was a young woman there named Kate Muther who had come in. She had all sorts of ideas about marketing and I didn't like them, and I drove her out of the company. That's my view of it. Stupidly, I might add. She went directly to VP of marketing of Cisco, and contributed mightily to their growth. There was a mistake right there. Then another mistake is: their router, because it was developed at Stanford, supported, I believe at one point, 14 different protocols. We supported TCP/IP and XNS and Applealk, as I recall. But they found 14 to support, and that was what succeeded. TCP/IP hadn't won yet. There was still a big jumble of protocols out there. They could go into a customer and say, "Which protocols do you want to use, or might you ever want to use ever? We've got 14 of them." That was appealing, because people wanted future-proof, and so they knew they could go in any one of 14 directions or some combination simultaneously. The IOS, Cisco's Internet Operating System, supported 14 protocols, none of which anyone ever ran other than TCP/IP, but having 14 was killer. So that's why-- I know I'm simplifying. Oh, I forgot the fact that they had excellent salespeople, and Kate Muther and John Chambers. They had excellent people, too. That might have something to do with it also. But they blew past us. Then when we botched the Bridge acquisition -- you know, we spent two years doing internal political stuff instead of growing the company -- that's when they really powered past us and became Cisco.

Shustek: If it wasn't for that internal problem in the acquisition of Bridge, did Bridge have the right product line to be able to compete with Cisco?

Metcalfe: Arguably. We'll never know for sure, but arguably. That's why we purchased them, or that's why we bought them, depending on how you put it. I mean, in the end of that story, by the way, we had wanted… [Chuckles] I wonder if I should say this? Yeah, I'll tell you this story. It's really interesting. So we acquire, or merge with, Bridge, and it had two prominent-- well, several prominent -- people, including Eric Benhamou, who ultimately became our CEO. But Bill Carrico and Judy Estrin had founded the company and done a great job, and we were happy to have them. We made Carrico president and, as I mentioned earlier, Krause’s plan was that Carrico would run the company ultimately. Judy, I think she became, I'm a little vague on this, but something like CTO, a prominent position. They were married, Judy and Bill. 3Com had from inception, -- well, from slightly after inception-- had been very careful about familial relationships among the employees. We had some bad experiences in the early days I won't go into. Oh, well Robyn and I did the first 3Com product together and we decided to stay married; she never worked at the company. We had rules about nepotism and stuff. So we were very careful that Bill and Judy would not be in the chain of command together, because they were married. This irritated Judy to no end. She says, "Look, I'm Judy Estrin. He's Bill Carrico. I don't want you doing anything special because we're married." And I'm saying, "But you are married, and we’ve learned that that complicates things, Judy." I remember having this conversation with her, and she was very offended, and I was very annoyed. But we arranged for them not to report. Actually, Bill Krause and Bill Carrico discovered that they didn't really get along. Carrico is a great startup guy, but 3Com was no longer a startup. We were now five times bigger than Bridge had been when he was running it. He wasn't really great at being the president of this much larger company. So friction developed and eventually Carrico resigned and left.

Shustek: At that time, when Carrico was president, what was Krause's role?

Metcalfe: I stopped being chairman, and he became chairman and chief executive. I became director and I was running the hardware division.

Shustek: Okay.

Metcalfe: So I gave him the -- he'd hate for me to say this -- I gave him the chairman's job and he gave Bill the president's job. But then Bill sort of flamed out as president -- Bill Carrico -- and resigned, under pressure, and then Bill [Krause] went back to being president and chief executive.

Shustek: What was Eric Benhamou doing in the meantime?

Metcalfe: He was running the software division.

Shustek: Okay.

Metcalfe: Competently. But anyway, to end this Carrico-Estrin story. These are both great people <inaudible>. The same day that Carrico quits, Judy resigns. I said, "Judy, I thought you just told me that you and Bill were like separate entities, and that we shouldn't pay any attention to the fact that you're married, and you're leaving on the same day, Judy. What does that tell us? It tells us that you really are married, and that it really does matter that you're married. Why? We want you to stay. Why are you leaving?" "Well, I don't like the way Bill is being treated and I'm going to go, too." So they went off and founded two or three more successful companies together, which is what they're really good at. Not running a big company like 3Com had become by the time they got there.

Shustek: Talk about Eric Benhamou and his accession to power. How did that go?

Metcalfe: Well, Eric's an engineer. Eric Benhamou. I think he, when we merged with Bridge -- I'm a little vague on this -- but I believe he was a prominent engineering manager. When I stopped running the software division and went over to do the workstation division, which was another exciting story, he became head of the software division and did a great job with it. Now, I think he may have done the ill-fated Microsoft deal, so maybe his record isn't completely flawless, but he was a very -- he is a very -- <inaudible> guy. So when it became time for Bill Krause not to be chief executive, there were three candidates: myself, running the biggest division of the company, Eric Benhamou, who was running the software division, and Bob Finocchio, who was running worldwide sales and marketing. The board chose Eric, and that's when I retired.

Shustek: Did Bob Finocchio stay around, or did he leave, too?

Metcalfe: He did. Yeah, successfully, for some time. Then later he became the CEO of Informix, so he left, but it was some years later. He didn't retire. He didn't take it badly that he wasn't chosen as CEO, but I realized I had to retire at that time. In fact, I should have left several years earlier.

Shustek: Why?

Metcalfe: I think I had become counterproductive. I was VP marketing, corporate marketing, in my last year. I think in retrospect, when I retired, I realized I probably would have been happier in the company, it probably would have been more successful, had I left earlier. Something about my mindset that wasn't quite appropriate.

Shustek: Interesting. Talk about the workstation business that 3com got into, and why.

Metcalfe: Well, we had servers and we had cards, and PCs had sort of been commoditized, and we noticed a lot of the expense and complication of these PCs was the floppy disks and the hard disks and the backup and the administration. So we decided we would build a DOS machine, or a Windows machine, both, that would be a network computer. That term later got picked up by Larry Ellison [of Oracle] and others and made famous; the idea of a PC on a desk that was designed to be on the network so it wouldn't be a standalone machine. It wouldn't have any disks, for example. All the disks would be on the servers. I picked up an idea from Steve Jobs and decided that our workstation shouldn't have a fan either. We decided to introduce our own network station that we would sell with our servers and our software and our card, into enterprises. I said, "I want that job," and they gave me the job, and one year later we were shipping a workstation. It cost $2,000. Priced at $2,000, list price. We built it for $1,000. It had no fans in it. It was a cool machine and I have one upstairs. I really loved it. An odd thing about the design is that the size of the box, which is quite small -- it was like a pizza box, sort of -- was determined entirely by the connectivity. We needed connectivity out the back, you know: the Ethernet adapter, we needed two kinds, outboard, inboard, the monitor, the keyboard. The back was so small that its size was determined by room for the connectors in the back. It had no fan and no disks so it was completely silent, which turned out to be a challenge because people expected their PC to make noises. But it did have a speaker. So we put noises into it so that when it was seeking on the server disk, we made disk noises. And it flashed a little light. Actually, I think it was later a light sufficed. You didn't actually have to make the noise, you just had to show disk activity. What would happen is the user would keep hitting a key trying to make it make noise, and all he needed was a little cue that the thing was actually seeking the disk and to be patient for another second or so. The first generation of this machine was very successful. We sold hundreds of thousands of them. Maybe 100,000, let's say. It became a big business for 3Com very quickly.

Shustek: This was running Windows or UNIX?

Metcalfe: DOS and Windows.

Shustek: DOS and Windows.

Metcalfe: Well, Windows was new. Windows came on right around the same time. It was a DOS machine, let's say. That's when I learned about dealing with Microsoft, because we had to get a DOS license and then a Windows license from Microsoft. It was a horrible experience. You could see that they had a monopoly and they were leveraging monopoly power, and they were extracting huge taxes as a result of this monopoly power. Then you saw, when Windows came, they demanded that we pay them a royalty for both DOS and Windows for every machine we shipped, whether or not it used DOS or Windows. And that we had to meet a certain quarterly minimum, whether or not we sold any machines at all. It became clear they were making all their money on minimums; no one was achieving their minimums, they would just pay. We were paying them quarterly minimums, a lot of money for software that wasn't being used.

Shustek: Presumably these were the same terms they were applying to other manufacturers as well?

Metcalfe: That's right. They were. Of course, it was harder for us because we were very small. We weren't Compaq or IBM. It was a really abusive situation. That's when I first caught onto the fact that, duh, monopolies can be abusive. We needed DOS or we didn't have a machine, so we had no negotiating power at all. Oh, and then there were alternatives to DOS, but if we used them, we had to pay Microsoft anyway. They negotiated that you had to pay them even if you didn't use their software. That's where, I think, they started going anticompetitive.

Shustek: Was there a UNIX market for this machine?

Metcalfe: No. It didn't run UNIX.

Shustek: Why not?

Metcalfe: It was a 286 [Intel 80286 microprocessor]. It couldn't run UNIX very well. It couldn't run anything very well, actually, <inaudible> DOS. Then we started the second generation of this machine. That's where we lost it, because the “second system syndrome” set in and we incompetently spent too much money developing a machine that came out too late that was not quite right. Then we killed the division. Actually, I think it was Benhamou who killed the division when he took over as CEO. He noticed we were in too many struggling businesses, and so he closed a few of them.

Shustek: But was it just because the second system was too ambitious? Was this a good idea, do you think? Were there other people trying to do it? You had sold hundreds of thousands of them.

Metcalfe: It'd be 100,000. This is highly debatable, and people who work there don't agree with me about this, but my view is it would have been a better… The reason that we ended up closing that business is because we did not execute. Here's the core of it. The first machine was developed under contract. The lead developer was an outside contractor named Bernard Daines, who's a very famous man. He's founded several companies since then, including World Wide Packets, I think, is one of his current companies. But anyway, Bernard was a consultant and had been a consultant to 3Com on previous projects. So I got him to be the developer -- the core developer, he was a chip and board developer -- to do that first board, and then there were 3Com engineers all around him. When we went to do the second machine, the 3Com engineering bureaucracy -- my bureaucracy -- said, "Well, it's not healthy that these machines are being developed by an outsider, so we're going to do this one ourselves, and we're going to get rid of Bernard." So we did. But the inside guys were not nearly as good as Bernard, so the machine got poorly specified, it took too long to finish it, when it was finished, it was too expensive -- because of the difference between Bernard and our internal people. It was my fault, because I let this happen. I was head of this division. I should have said, "Well, Bernard just performed magnificently on our first machine. Duh. Why don't we let him do our second machine, since that one…" Instead it was this sort of NIH [not invented here], bring it inside. Why should he have all the fun? We want to design the machine and blah, blah, blah. So we ended up incompetently doing the second machine.

Shustek: But if this was a good idea in an expanding market area, why wouldn't other competitors have done similar products?

Metcalfe: Well, they were talking about it, and there were things. I think even Novell talked about a diskless machine, too. We weren't alone. But we had, by far, the best machine. The 3Station, it was called. Another beauty of this design was we realized, being in the cabling business, that cables were a problem. So we designed the machine so that it was raised up about this high and had channels through it so the keyboard cable and the monitor cable and all the cables were hidden elegantly through these channels we had cut in this box. It was a beautiful little thing. Even when the cables were installed, it still looked good. The mouse cable came out of a little hole, and the keyboard cable. We had both right-handed and left-handed channels, depending on whether you were right-handed or left-handed. We did the fan thing right, and we did the cabling thing right, and we did the Ethernet thing right. And it had great administrative advantages in that you didn't back up your machine. You didn't have to be sure all your disks were backed [up]; you didn't have any more disks. You didn't have that little box of floppy disks anymore that everyone used to have in those days, because everything was on the server, which is sort of how it should be.

Shustek: Was this still a disk sharing system, or was it file sharing? What was the interaction between 3Stations?

Metcalfe: It ran both NetWare and our network operating system, and so it was both. And our disk sharing software by this time had become file sharing. Too late to unseat Novell, but… You'd just turn on the machine, it would boot up over the network and you'd have a DOS machine there, and you could run applications. And it was administered centrally.

Shustek: What was the performance like?

Metcalfe: It was, in some ways, better than the PC because we put a lot of memory on it and we had RAM disk. There was no “disk” disk, but there was a RAM disk in the machine. In a lot of applications it was just faster because it would just access a RAM disk instead of going all the way to the… Or a hard disk on the server instead of a floppy on the desktop, so it would get much better performance by going to a shared hard disk than a local floppy. There was no more floppy. A lot of people would whine, "Oh, we want a floppy. We want floppy disks." Remember, they used to complain that Macintosh didn't have floppy disks, either. Well, floppy disks were a thing of the past. In fact, they are a thing of the past, actually. <laughs> We were just among the first companies to sell them. I remember in the shop at the Exploratorium in San Francisco, for years afterwards I would go there and there were the 3Stations in the shop running the cash register. For years! You know how private institutions like that don't have a lot of money, so when someone donates equipment it just stays there forever. I used to sometimes go out of my way to go to the Exploratorium, just so I could visit the 3Stations in the shop there. I guess eventually they replaced them.

Shustek: The notion of how much computation is in front of the person versus how much is remote is a pendulum that's swung back and forth many times. In some sense, 3Station is a move back toward timesharing or external type interaction where the user interface is in front of the user and the rest of the processing-- or some of it, at least-- is centralized.

Metcalfe: Well, be careful there. The swing back wasn't as extreme as you just portrayed it. Windows and DOS and Word still ran on your desktop. It was just the backend data functions that were…. Yes, it was a slight swing back, but it was not as extreme. We were still using the microprocessor on the desktop to do that high interactivity that is the advantage.

Shustek: Do you think that this didn't succeed as a computational model because disks got so dense and so cheap so quickly?

Metcalfe: Well, I think it has succeeded as a computational model, by degrees. That is, PCs in corporations now to a large extent rely on servers. And the backups are automatic. Now they do have local disks, I suppose. But we had local disks, too. Ours were made of RAM. Theirs are made -- the modern ones are made -- of multi-gigabyte disks. So it's hard to say. But you're right, it is a pendulum. But in a way, it's many pendulums. You know, the disk storage swings back, the processing swings back and forth. You have compute servers. How much of the stuff you do on your machine now is done on the Internet? You know, like Google isn't even done in your building anymore. It's just you have a little Windows interface and you're using Google all day.

Shustek: A lot of the applications are starting now to run on the servers on the Internet.

Metcalfe: Amazon is an application that does not run in your building. And on it goes.

Shustek: Yeah.

Metcalfe: So yeah, there's lots of little pendulums swinging back and forth.

Shustek: Before we leave 3Com and go onto the next phase, can you talk a bit more about some of the other Ethernet competitors that have managed to continue on? Storage Area Networks tend to use Fiber Channel. Why didn't this all converge, in the interests of compatibility, to Ethernet?

Metcalfe: That's funny. The way you ask that question is the opposite premise to the one I have in my head. Ethernet has, over its 34 years, killed a long series of competitors. It's visibly killing Fiber Channel and InfiniBand right now, and SONET is doomed. So the wars continue and Ethernet continues winning. Well, what happened to FDDI? And what happened to Token-Ring? And what happened to ARCNET? And what happened-- what are some of the-- ATM? What happened to ATM? They're all being killed by Ethernet.

Shustek: Why do people keep trying?

Metcalfe: Well, it's this pendulum thing. Well, there's also subtlety beneath this. There's the pendulum thing. You keep saying "Well, Ethernet's good for that, but we have to do this, and this is very specialized; like Fiber Channel for storage. So we're going to…" And there are lots. There's FireWire, and USB, and some of which are quite important because they're so specialized, Ethernet’s specialized. One of the reasons that Ethernet has been persisting for 34 years is this funny thing has evolved where, when a new technology comes along, they call it Ethernet. It isn't as if the thing that Dave Boggs and I built in 1973 has been winning for 34 years. It's just that every time something wins, they call it Ethernet. The things that are called Ethernet today are quite different and diverse. But Fiber Channel and InfiniBand are doomed, and SONET is doomed. T1 is doomed. But, of course, when something's doomed, especially if it's a networking technology, it takes a really long time to die.

Shustek: What about USB?

Metcalfe: It's great. I use it all the time. There are reasons for diversity of standards. There's considerable diversity within Ethernet. The generations of Ethernet coexist. In fact, my Mac has a single plug, and it decides whether it's 10, 100 or a gigabit per second. It's a miracle, to me. Just depending on what switch I have. It just decides. What geniuses came up with that? There's variety as technological advance disrupts the standard. Then there's variety because of different uses, and USB is a great example. For what USB does, that particular physical interconnect is superior to the RJ45 that we use predominantly in Ethernet.

Shustek: It can't just be the physical interconnect, though, because Ethernet could adapt to have an alternate physical connector, if that was the issue.

Metcalfe: Exactly. And it's conceivable they could have called USB “Ethernet”, but they didn't. They happened to call it.. Well, WiFi was originally called wireless Ethernet. Now they call it WiFi. There's a case where the brand name, the Ethernet brand name, has been shunted aside. It is wireless Ethernet. You look at the packet formats [and] it's very close to the 802 packet format. But they've decided now… So that's a departure. That's a blow. That means Ethernet is maybe peaking now as a name. Here's wireless Ethernet, becomes WiFi. Hmmm, that's interesting. Yes, one could imagine building an RJ45 with a little flash memory attached to it that you just plug into an Ethernet adapter and it does exactly what a USB, and is… I don't know why. There's probably some simplifying circuitry, some optimization or just an accident of history. I'll have to remember to suggest to the USB people that they call it Ethernet something. Pluggable Ethernet. Flash Ethernet. Ethernet Flash. EtherFlash. Call it EtherFlash. Yeah.

Shustek: Is the Ethernet name trademarked? Does somebody own it, or is somebody responsible for its use?

Metcalfe: I am responsible for its use. In 1973, on May 22nd in a memo that I wrote at the Xerox Palo Alto research, I coined the word "Ether Net." It used to be two words, with a capital N. Then it became one word with a capital N, and then we dropped the capital N and now it's capital E, small N. It was a Xerox trademark briefly. Then when we brought Ethernet to the IEEE -- when we formed the DIX consortium, Intel and DEC said, "We're happy to join with you in this consortium, but we can't use the name that you've chosen." Xerox thought Ethernet was kind of a nerdy name, so they had changed its name to “The Xerox Wire”. The Xerox Wire, a great name. DEC and Intel said, "We'd love to work with you on it, but we're not going to call it The Xerox Wire. Why don't we call it [slaps his head] Ethernet?" Then they went to the IEEE. The IEEE said, "Well, we're going to call it “802.3”. We really don't want to use the word Ethernet, because that's a trademark of Xerox Corporation." Then Xerox decided to let go of the trademark. So it's really an uncontrolled word. Who's in charge of that word now? It has to be me. I can't think of anyone better. So I'm in charge. And this came up. HP introduced 100 megabit per second technology in the 90s that they tried to call “Fast Ethernet”. I noticed that it was sufficiently different from Ethernet in its operations that I didn't think it should be called Ethernet. I was then a journalist, so I wrote a column in InfoWorld attacking HP for trying to use the word Ethernet for this kludge. A big fight ensued, and then eventually they had to change the name. They changed it to “100VG-AnyLAN” became its new marketing name. Have you heard of, have you bought any recently, of 100VG-AnyLAN? No, I'm sorry. It's called Fast Ethernet. It runs at 100 megabits per second. Now, of course, it's passé because we've moved onto gigabit Ethernet. “GE”, gigabit Ethernet.

Shustek: You left 3Com in 1990. Did you have any continued involvement with the company after that?

Metcalfe: Yes. It is pretty normal. But when I left, when I retired, I remained a consultant for another year, a paid consultant to the company, during which time I went to England as a visiting fellow at the University of Cambridge in England. That was my first attempt at retirement. That didn't last, so I came back to be publisher of InfoWorld.

Shustek: We'll talk about that in a second. But after that year after you left, you had no further involvement in the company. Do you have any observations on what happened to it after that?

Metcalfe: When I left, the company was around a half a billion a year in revenue. Next thing I knew, it had $5 billion a year in revenue. People began to tell me that I was the inventor of the Palm. People would show me their Palm, and it had 3Com written on it, and they knew I had founded 3Com, so they assumed -- a lot of people assumed -- that I had invented the Palm. The hilarity of that was extreme. First of all, I wasn't at 3Com when 3Com bought U.S. Robotics, or merged with U.S. Robotics, a horrible mistake. No one knew that when 3Com did that, they had no idea that Palm was part of U.S. Robotics, that it was just buried in some division somewhere and that they had acquired it accidentally. Then no one knew that 3Com had sort of botched the care and feeding of Palm, so that its founders had to leave and start a whole new company. They just naturally assumed that since 3Com was written on the Palm, that I had invented it. Of course, I was not quick to deny it. <laughs>

Shustek: Why was the acquisition of U.S. Robotics a mistake?

Metcalfe: Because U.S. Robotics was a fraud and had been stuffing its channel and had outmoded products that no one wanted to buy, and 3Com stupidly bought it. The mistake was slightly more conceptual. Eric Benhamou, after he took over as CEO, did a series of acquisitions, all of which were successful, prominently so, and a big break from the series of unsuccessful acquisitions 3Com had done prior to his becoming CEO. The reason is, he had learned from the unfortunate Bridge experience, where he watched 3Com and Bridge try to merge. So he adopted the principle, explicitly, that 3Com was never going to merge with anybody. It was going to acquire much smaller companies. That was a formula that worked. We are buying you. We're big, you're small, we're buying you, and no, you're not the president anymore, and you're not the head of Germany anymore. We already have one of those. And they would just efficiently, on the first day, let everyone know what was going to happen instead of futzing around for two years figuring it out. U.S. Robotics and 3Com were about the same size when he decided to merge the two companies. That was his mistake. These two, two and a half billion dollar companies, roughly, merged. That was the conceptual error. He broke his own rule after a string of great successes. Plus, U.S. Robotics was, I hate to say it, but, a fraud. Modems were going like this [a curve going down] and they had been grossly stuffing the channel So there was this big magilla [that] followed right after the merger where they discovered that a lot of the sales were fake.

Shustek: Why do you think Eric did that? Did he not know U.S. Robotics sufficiently well at the time of the merger?

Metcalfe: I wasn't there and I don’t really know what happened, so I'm really making stuff up here. But he just got snookered. It happens all the time.

Shustek: Do you think that was a cause of what's happened to 3Com since then? How do you rate 3Com in the years since?

Metcalfe: I have very little knowledge of 3Com since I left in 1990. I did notice that it grew to five billion, which was a lot bigger than when I was there. So the board of directors had made the decision again of CEO correctly. Eric did great. The Bridge merger is what caused Cisco to exist. I guess the U.S. Robotics [merger] sort of broke the back of the momentum that 3Com had going into the bubble, so that's why Cisco's even bigger, and 3Com is now less, I guess revenues must be below a billion now. Probably three quarters of a billion.

Shustek: Yeah, and it gave them Palm.

Metcalfe: Well, there was this funny period in which Palm, which 3Com owned what, 95 percent of, if not all of it, and the market cap of Palm was bigger by far than the market cap of 3Com, which was weird. I kept asking people, "How can that be? What sort of Wall Street math allows that to be true?" No one has ever adequately explained that to me. But I guess, then they spun off Palm and gave their shareholders a great benefit at that. Part of the shrinkage had to do with spinning Palm off, and part of it had to do with just, when the bubble burst, a lot of companies went down. 3Com went down with them. Some of it had to do with the company having lost its way somewhere along the line. Now the new CEO, Edgar Masri, he and I worked together at 3Com, and then he went off to become a venture capitalist in my same building here in Waltham, and then he's gone back now to take the challenge of making the most of 3Com.

Shustek: Which is now an east coast company.

Metcalfe: That's another irony. When I founded 3Com on the phone from an apartment I had on Beacon Hill not far from here, I founded it as a California company because I was also living in Palo Alto. I had an apartment in Palo Alto and one here. So for years afterwards, whenever I was in Massachusetts, I would tell people from Massachusetts that 3Com had been founded in Massachusetts, and then when I was back in California, which was most of the time, I would tell people it was founded in Palo Alto, both of which were actually true. The headquarters was moved around from Palo Alto to Menlo Park to Mountain View to Santa Clara. I guess Santa Clara. Then it moved to Marlborough, Massachusetts. So now people think 3Com was founded in Massachusetts and has always been here, which is not true. I mean, it's always been here a little, but now it's headquartered here in Marlborough. I think it's still Marlborough. So that's a bit of irony. But I don't know anything about 3Com. My contact has been very distant. I've been gone longer from 3Com than I was there, so it's really quite a remote... So every time something bad happens at 3Com, I'm not there anymore. And every time something good happens there, it's “I founded the company.” Very situational.

Shustek: Great.

Shustek: You once said that your life, so far, had four careers; as engineer, entrepreneur, pundit and VC. Let's move into the pundit phase. After you left 3com, you were a consultant for a while. Why didn't you start more companies? Other people have become serial entrepreneurs, and you didn't.

Metcalfe: Right. And I'd considered it. In fact, really did start a couple more companies, but only from a distance. For example, Grand Junction Networks, which is the company that introduced Fast Ethernet, I was a founding director of it and worked with Howard. I don't know, I'll probably annoy the people associated with it, but I convinced Howard Charney to start that company. In fact, Howard came to me and said, "Bob, let's start a company, like we did 3Com." Howard Charney. If you were ever going to do anything in life, if you can get Howard Charney to do it with you, [signals approval]. He came to me and said, "Let's start a company." I said, "No, I'm not going to start a company." I'll get around to answering why not. I said to Howard, "Why don't you start the company?" He said, "Well no, you talk the talk, and walk the walk, and have the network. I'm the engineering guy and the manufacturing guy. You should really start the company." I said, "Howard, you could do it." And he did. He started Grand Junction Networks and sold it to Cisco for 2, 3, $400 million. That was fast Ethernet. I was a founding investor there, and was in on the early dinette table discussions of what to do, but ultimately did not want to start a company because I burned out at 3Com. That is, those 13 years worked out very well, but they were hard, and especially the last part, and I didn't want to do it again. I felt I could sort of stay in the same community and just take a different role. The beauty of my career at 3Com is I got to do a lot of different things. I got to run divisions. I got to run sales. I got to run marketing. I got to do PR. I got to do advertising. I got to do engineering design. I got to write programs. I wrote programs for money at 3Com, briefly. That didn't last very long, but I did. I wrote an Epson printer emulator in postscript, and we sold it to Xerox for money. I wrote it, myself, and then we decided that was not a good use of my talents and that I should go run marketing instead or something. Maybe it's a variety thing. That is, I did that already; now I want to do something else. I went into journalism. I was a publisher/journalist for ten years, and then for the last six years, I've been a venture capitalist.

Shustek: Had you any previous writing experience? Why did you think you could be a journalist?

Metcalfe: One of the things that appealed to me about journalism is that if I were a journalist, I wouldn't have to do performance appraisals anymore. I believe June of '76, I started having people reporting to me at Xerox. I had to start doing performance appraisals, because they're very important and they needed to be done well. I just never liked doing them. I've done a lot of them and I still do them. But being a journalist, you're an individual contributor and you don't do performance appraisals and I really liked that part.

Shustek: Why is doing a performance appraisal so painful?

Metcalfe: It's important. It must be done, so it's not as if I have the option of not doing performance appraisals. There are people who adopt that option, and they're being irresponsible and ineffective. [Muses] Why don't I like it? Because I do performance appraisals every day. That is, I prefer, as I talk with people, to tell them how they stand on what they're doing and how to improve every minute of every conversation. Stopping to do it in writing on an annual or semiannual basis seems redundant, and reducing it to writing and then having what is often a difficult discussion. I'd much rather have it a little bit every day, than all at once every year. By the time I'm done with you, you know exactly what your performance is in my opinion, and you don't need me to write it down. But then I do write it down and then we have it out again. I don't like that part.

Shustek: What were your steps in becoming a journalist?

Metcalfe: It happened sort of by accident. I do like to write. I believe part of my success package is writing. The demos I wrote at Xerox. The Ethernet memo that I wrote in 1976 is a nice piece of writing. I'm really proud of it. I must have spent two years writing it. Xerox wouldn't let me publish it. This Ethernet paper became a CACM [Communications of the ACM] paper in July, '76. I really had to write it for a really long time, because they wouldn't let me publish it, so I kept iterating it. It's a pretty good piece of writing. I admire writing and writers. Oh! My first exposure was at MIT. My advisor for a while was Jay W. Forrester, the inventor of core memory and builder of the Whirlwind computer and so on. He was my advisor. There were 12 of us. He advised a group of 12 of us, the undergraduate systems program in the Sloan School of Management. I was one of 12 and he was our advisor. One day, Jay says, "By the way, I'm going to run a writing seminar, if you guys are interested." He said, "I'd like you to come to my office every Friday and we're going to learn how to write together." The inventor of core memory wants to teach me how to write. So a group of us-- not all 12 of us-- went to his office and he gave us our first writing assignment. I forget what it was, ironically; write an essay about something. So we all went off, came back the next Friday with our essays. We read them to each other and commented on them, and he said, "Okay, I'd like you to write it again, same topic." My recollection is he asked us to write 19 times the same essay. Did I learn to write from Jay Forrester? No. What I learned from Jay Forrester is that Jay Forrester thinks that writing is important, so then I thought writing was important, and so then I began writing a lot. So my business plans, and my sales presentations, and my corporate memos were all written, more so than your average engineer or sales person. I like to encourage sales people and engineers to write, because it leverages them. Once you write something, it has much more power than if you just blab it.

Shustek: Before becoming a journalist, officially, had you written articles for trade journals or newspapers?

Metcalfe: I'd given a lot of speeches, a lot of which were written. I had written papers, academic papers, not many, but a series of them. I remember writing my Ph.D. dissertation was very hard, so writing does not come easy to me.

Shustek: And you have written RFC's [Requests For Comments].

Metcalfe: I had written RFC's and I enjoyed writing them. That's right. Thank you for reminding me of that. I've always liked writing as a way of getting leverage, because you say it once and then it gets read a bunch of times in places you didn't expect. So yeah, the power of writing as a leverage tool; writing and speaking, I would say, writing, speaking, presenting is a way of leveraging, selling what you're doing. When I retired from 3Com, I was invited to be a visiting fellow at Cambridge University, and I was also invited to write pieces about Ethernet; journalistic pieces in trade publications, generally. As I was going to Cambridge, the editor of *Communications Week*, a weekly trade rag, said, "I'd like you to write a weekly column for me." I said, "Well I'm going to Cambridge, England." So we decided that I would write about what was happening in Cambridge, England in the computer laboratory there. I started writing a weekly column and that was my transition from freelance to staff. I had a regular weekly deadline, and I really liked that. I liked it because generally my method was to keep ten columns in progress at all times, up in the air, working on them. Then my deadline would approach and I would choose one of them and write it. Then I noticed that the people around me, grad students at the computer laboratory, suddenly wanted to be with me, because they wanted me to write about them. So they all became little flacks, and I became the hack. I noticed I had access to people who would talk to me about what they were doing because they wanted me to write about them, and that gave me a way to learn about what they were doing. It was a very useful tool for satiating my curiosity about things. If I got curious about something, I'd just take it up as a column project and then I learned about it. I did that for about a year, and then got [Pat] McGovern, who runs IDG, International Data Group, which publishes 300 magazines around the world, invited me to be Publisher/CEO of *InfoWorld*, which was at that time a $40 or $50 million publication, one of the top ones in the world. I said, "Wow, I could start at the top: CEO of a big publishing company."

Shustek: Is that a writing gig or is that a management job?

Metcalfe: That's a management and selling job, selling advertising. For two-and-a-half or three years -- I forget, something like that -- I was publisher/CEO and I learned how to sell advertising. I sold hundreds of millions of dollars of advertising, which is an incredible world that I loved learning about: how to build an audience of readers and how to sell them to advertisers. That whole business is a huge business and it's very subtle. It's even more complicated than high-speed network interfaces. I learned how to sell advertising. It was great. But then the editor in chief, Stuart Alsop, invited me to write a publisher's column. A publisher's column is a very standard thing and you generally don't read them. I started writing a publisher's column. But I had been writing this column at *Communications Week*, which I had to stop because I had found a new job. So I started writing a column that wasn't quite a publisher's column, and one day, this wonderful thing happens. Stuart Alsop came into my office and said, "Bob, I want to remove the world publisher from your byline. I'd like it just to be your column, "From the Ether", with your byline. It won't be a publisher's column; it'll just be a column." That was a promotion, right, because the publisher's column is kind of perfunctory rubbish, generally. So then I became a columnist. This took a year or a couple of years. The problem was that as a columnist, I loved to opinionate and be nasty. Tell the truth to my readers. Be interesting. That sometimes involved attacking vendors of equipment to whom, with my other hat, I was selling advertising and marketing programs to. Eventually that became untenable. Too many of my customers said, "You can't write that about me and sell me ads at the same time. You're not allowed to do that. Your magazine can do it, but you can't do it." That's when I moved to Boston, stopped being publisher of *InfoWorld,* and became IDG's vice-president of technology, which meant pundit. Sort of a corporate level pundit writing for *InfoWorld* and organizing conferences and working with *ComputerWorld*. There we sold a billion dollars of advertising every year.

Shustek: Did you find that satisfying? As satisfying as creating companies and products?

Metcalfe: I did it for ten years. I loved every minute of it. Not every minute of it. You never love ever minute of everything, but it was great. I took my best columns and I published a book. IDG published my book, *Internet Collapses and Other InfoWorld Punditry*. It's still available on Amazon.com. You can buy it used for under $2. I had an illustrious career as a pundit. I predicted the collapse of the Internet. I predicted the bursting of the Internet bubble. 50%; I was half right.

Shustek: Let's talk about some of your predictions. One of them has become famous, as "Metcalfe's Law". I don't know when it first became published. I know that George Gilder talked about it in a *Forbes* article in 1993. What is it and when did it first appear?

Metcalfe: Metcalfe's Law. Well I like the fact that it's called Metcalfe's Law. Often, they talk about Moore's Law and Metcalfe's Law. I love when they do that, because being associated with Gordon Moore and his law is just really social climbing for me. In the early '80's, I was selling Ethernet. It went like this. We developed an EtherLink for PCs. We have a sales force, now 12 people, and we're a tiny little company. We made a rule: we can't call on people who don't already have PCs, because it's IBM's job to convince them to buy PCs. It's our job to convince them to use EtherLinks to network them. Let's not evangelize PCs. Two is; don't call on anyone who has Apple II's. We'll let Nestar do that. Ethernet is too powerful for the pitiful Apple II and we'll just focus on the IBM PCs, a very wise business decision. Third; all of our sales people used to sell minicomputers and they cost about $30,000. So we're going to try to sell 30 node networks so that we'll get $30,000 per sale, and then we'll all get the right amount of commission and then we can send our children to college. Well we learned that the first two qualifiers worked. That is, it was really shrewd to not be evangelizing PCs, because IBM was doing that, and it was really shrewd to not be doing Apple II's because that would have divided our engineering and support efforts, and we let Nestar have that business. But the $30,000 thing was a problem, because LAN's were generally unknown, and printer sharing and disk sharing were interesting, but $30,000 to try something I never heard of? Our sales cycle was infinite; we weren't getting any business. We decided -- it was in our offsite in Lake Tahoe. We rented a house. We had an offsite there. One of the revelations of this offsite is we were going to sell three node trial networks. We put together the promotional materials and we told our sales people, "No more $30,000 networks. Don't do that anymore. Stop. We're going to sell trials. I know your commission on that's pitiful, but there will be follow-on business and that's where you'll make your real money." I decided that. I was head of sales and I enforced it on our sales force. And it worked. Suddenly, our sales cycle collapsed to one week, because people could spend $3,000 a lot easier than $30,000, and they did. The idea of printer sharing and disk sharing was enough to sell a $3,000 trial. Our products worked, which was a plus, but our customers said, "Your products work just like you said, but they're not very useful." Well that's devastating. Why aren't they useful? Well the answer is a three node network can't be useful. It's too small. Hmm. Why is that? So I do a graph. The graph had number of nodes on the network along the bottom and dollars vertical. And the cost of the network was a straight line. It went up a thousand dollars per node like this <whizzing noise>. But what's the value of the network? Well, the value of the network must have something to do with the number of nodes that you can connect to from your PC. And then each node has that value, that is, it can talk to N-1 other nodes, and there's N of those. The total value of the whole network is N times N-1, which is approximately N squared. That becomes a quadratic line. It goes like this [shows steeply increasing line]. There's this point out there where the linear and the quadratic cross. For small N, the value is below the cost, and then there's a critical mass point, and then above that the value greatly exceeds the cost and gets better all the time. Network effect. I made the slide, gave it to the sales force and told the story: the reason that your three node networks are not as useful as you like is that they're below critical mass. The way that you begin to experience value is to buy more of them from me. I'm not exactly sure where the critical mass point is, but I'm pretty sure it's above three, so let's do that. So we went back to our customers. The products work; it's just they're not very useful. We gave them a reason why they weren't useful; you have not achieved critical mass. That reason seemed plausible. Wouldn't you know, we started selling $30,000 networks to these people. They believed that story and thank God, the story turned out to be true. That is, when they bought the 30 node networks, they were useful. They accessed the Internet with them, which is a new capability. Oh yeah, the LAN email is much more useful when you could include a whole group, instead of just three. The LAN email was kind of a product no one was interested in, but they got interested in it once they started using it. No one ever imagined that you'd want to send email to somebody down the hall, but then as soon as they did it, they began to enjoy the benefit of it. It caught on, and it became much more valuable at 30 than it was at three, so that was a slide. And then in 1993 -- this is 13 years later or ten years later -- George Gilder started working on a book that ultimately was named *Telecosm*. It was a follow on to his earlier book called *Microcosm*. The book *Microcosm*, I believe, is the book in which Moore's Law was touted, came of age, escaped Intel. George was working on *Telecosm* and he interviewed me, and he asked me for a bunch of stuff. I showed him my artifacts and he saw this slide showing the straight line and the quadratic with the critical mass point. He said, "That's Metcalfe's Law." He wrote this article in *Forbes ASAP* in 1973 [sic] referring to Metcalf's Law: the value of a network rose as the square of the number of attached users, devices, whatever. That was '93, so Metcalfe's Law gets some usage, thanks to [him]. Then his book came out in '95, which gave it another little boost in peoples' consciousness. Then Al Gore started promoting the information superhighway. Al Gore -- let's not get into this -- Al Gore and I may not have invented the Internet, but we invented the Internet bubble. Al Gore started talking about the information superhighway, and then in 1995 or 1996 he shows up at the MIT commencement. I'm a trustee of MIT, I'm sitting on the stage, and he starts talking about Metcalfe's Law. Two years later Bill Clinton shows up at MIT commencement, and he starts talking about Metcalfe's Law. So Metcalfe's Law became one of the inflators of the Internet bubble, like the information superhighway metaphor. The notion of this value creation by connecting things together became part of the Internet bubble inflator. Then all these business plans got written referring to Metcalfe's Law and why you should invest in this company, and the bubble blossomed and you know what happened. And then it burst. Well, Metcalfe's Law is still out there. Now there's a new bubble inflating -- I'm not exactly sure what it is, the social networking bubble -- but Metcalfe's Law has begun appearing on PowerPoints again. An article was written in the *IEEE Spectrum* a few months ago, in which three professors ganged up on me. They wrote an article, a cover story, I guess, on *Spectrum* about Metcalfe's Law. Not only did they claim for the 19th, 20th time that Metcalfe's Law is wrong, they actually wrote that Metcalfe's Law is dangerous. They used the word "dangerous", because it's helping to inflate the social networking bubble. I've written a rebuttal to their article, which you can find on the Internet by blogging. It's a blog entry. But basically, in that I point out that these guys, after attacking my law and calling it dangerous, the only improvement that they suggest was that the value of a network doesn't grow as the square of the number of users; it grows as N log N. N log N is approximately N times N. That is, they never really got to the bottom of it. They're just refining a thing that's just conceptual to begin with. The proof of this is, of course, that no one has ever tried to assess the constants of proportionality of the law. No one's ever actually taken my law and tried to apply it to actual numbers, like they do to Moore's Law. Moore has data points that his curve sort of fits. No one, including me, has ever tried to put data against the law. And these three professors, they didn't do it, either. They just conceptually argued that N log N was better than N squared. Whoop-de-do.

Shustek: There's someone else, David Reed, who claims that it's 2 to the N.

Metcalfe: Yeah. David Reed, who I know and admire, has Reed's Law. I always rib him about it, because no one's ever heard of Reed's Law. <laugh> I mean very few people have heard of Metcalfe's Law, but no one has ever heard of Reed's Law. I like to rib him about it. Whether 2N, or N log N, they're all missing the point. The whole point is networking is valuable. That's the point.

Shustek: Yes, but in comparing it to something like Moore's Law, where in fact, you can plot specific numbers against the chart, what's the metric for value that allows you to represent it as mathematically as you did on the chart?

Metcalfe: There are flaws even deeper than that. There's no real notion of connectivity. You really need to know what it costs. My quantification of connectivity was the cost of the network. Then the value -- I never really defined what the value is, so the quantification's impossible. Well, I'm working on that. I'm actually trying to take another look at the law and try to quantify, try to do a Moore's Law kind of thing with some real data and try to assess what the constant of proportionality is, and see if the law is actually true or not. Who knows? It's probably not exactly true. In fact, there's a lot of evidence that it must peak out at some point and roll over. That is, there's six billion people on earth. Do you have anything to say to the last five billion? Probably not. On the other hand, as I wrote in my blog response to this attack in IEEE Spectrum, Metcalfe's Law recurses. That is, there's a value of being on the Internet, but then there's a value of being in the little networks that form on top of the Internet, like social networks, or the networks that form around books from Amazon. That's a network there. Blogs are a network, too. So I argue there's a recursion of Metcalf's Law at the higher level networks like this [shows a structure going up], and that if you integrate the total value of the base level plus all the recursions, it's N squared again, even though there's a roll off on the base level value.

Shustek: It's sort of like a fractal curve. Every time you look in microcosm, it seems like the whole.

Metcalfe: It's fractal. I regret now that I didn't include the word "fractal" in the title of that post. The post was, "Metcalfe's Law recurses down the long tail of social networking", and I should have said "recurses fractally down the long tail of social networking." <laugh> I should have said that. Thanks for that. I'm going to use it in the future.

Shustek: Metcalfe's Law has been a wonderful catalyst for lots of discussion about the value of networking, so in that sense, its extremely useful, regardless of how specific it is.

Metcalfe: Just for the record, I wrote a column about Metcalfe's Law a long time ago in which I made it clear that it's a conceptual thing, that I don't really think it's that serious a mathematical thing, that Moore's Law really is a different kind of thing that has a life and actual data, as opposed to concept. All this disclaimer. But then every week, some idiot out there discovers Metcalfe's Law and attacks me, because he thinks that I think it's exactly correct and that I take it all that seriously. Then I have to refer him, this poor schmuck, back to <laugh> my column that I wrote ten years ago agreeing with him. That's the wonder of the Internet.

Shustek: Let's talk about seriousness with respect to predictions. You made one in December of 1995 that had to do with the collapse of the Internet. What was the prediction, and were you serious?

Metcalfe: It was a brilliant success, actually, that prediction. I was a columnist at the time, with about a million readers, if you count international and pass-along. Actually, in the U.S., my readership was assessed by third parties at 629,000, but I'll say a million for round numbers. When I wrote my column every Sunday night, I wrote 615 words that I knew very shortly were going to be read by a million people. I loved that power. Every little word —this word, "of", "the", "and" -- is going to be read by a million people in a few days. Should it be there? Should I delete it? I like to delete articles, by the way. "And", "the"; they seem like busy words to me. It was a real lovely, powerful thing. But I wanted to succeed as a columnist. I wanted that million to be two million someday. That raises certain ethical questions about the extent -- and I believe many columnists make the wrong decisions ethically at this point, because there's a difference between being interesting and being right. You kind of owe it to your readers to be right, but you won't have any readers unless you're interesting. Anyway, at that time, I was learning to be interesting, because I wanted to get to two million readers from one million. I saw a number of problems with the Internet in '95, all of which were true insights: the fragility of the thing, the fact that Cisco routers could all be crashed at the same time, the fact that there were only twelve name servers, the fact that spam was taking off, the fact that no one had anticipated viruses and security, the fact that packet loss was going up and up and up because the traffic was exceeding certain circuits in certain places. I accumulated a long list of all these reasons which spelled doom to the Internet. One night, writing the column, <laughs> I decided how to phrase this. So I predicted. I had to write a column every week, you understand. I began to learn that my readers appreciated predictions, because they kind of relied on me to help them assess the future, and so on. How could I make this into a prediction that would be interesting? So I predicted the collapse of the Internet -- I chose that word carefully -- the collapse of the Internet during 1996. It was a bunch of columns. They're all in my book, by the way. My book is titled *Internet Collapses and Other InfoWorld Punditry,* available at Amazon.com. [It's] on the long tail. It used to have a two-digit Amazon rank. It now has a seven-digit Amazon rank, I'm proud to say, last time I checked, and I check often. But then I quantified this. I said that during 1996, there would be an outage of the Internet which exceeded one billion lost user hours. I called it a "gigalapse". I predicted there would be a gigalapse in 1996, and that's what I meant. Then I wrote a number of spoofy articles. I predicted that pornography would bring down the Internet. Not the actual pornography; it was all the writing about the pornography would clog the Internet -- there wasn't really much pornography at all -- and a lot of tongue-in-cheek and sarcasm. My editors always said don't be sarcastic. They'll rub your nose in it. I ignored that advice and wrote a lot of stuff. During 1996, there were some pretty big outages. The biggest one occurred in August of that year and it was, I forget the exact number, 118 megalapse. About one-eighth of a gigalapse. That's as close as I got.

Shustek: How do you measure a lost user hour?

Metcalfe: How many users were denied access to the Internet for how long? You multiply the number of hours times the number of users. This particular outage was a bunch of Cisco routers at AOL [that] came down. And the funny thing was -- and I was right about it, I wrote about this -- there was no genetic diversity. They were all Cisco routers, so the bug in one router was in all of the routers, right? The router went down. They reloaded it. But by the time they reloaded it, this one had gone down. It had somehow rebooted itself and so they went to fix that one and these two had gone down. They spent a whole day, 19 hours, actually, trying to bring these routers back up, but they kept bringing each other down faster than they could bring them up. Eventually, they got it under control. But before they had done that, a 118 megalapse had occurred. I trumpeted this in my column, of course, because it meant that, hey, this is only August. We still have September, October, November and December. <laugh> It could happen. But the year came and went and that was the biggest outage of the year. By the way, these columns generated all sorts of interest and activity. A lot of stuff got written. All the Internet intelligentsia were on my case. The problem was, I had been an Internet guy and they were used to being attacked from outside. I was attacking them from inside, so that really struck home. There was a lot of vituperation. In fact, if you see the cover of my book, you'll see I have a blurb from Vint Cerf on my book in which he refers to me as a "ranting gas bag". What he didn't realize is that I would put that on my book. <laugh> I asked him for the blurb. Vint Cerf, "father of the Internet". He sent me back this nasty blurb, assuming I wouldn't put it there, ha ha, but I did. So there he is on the cover of my book referring to me as a ranting gas bag, because of my accusations about his baby being fragile and on the verge of collapse. Basically, everything I said, except the actual numerical prediction, is true. I think the service I performed is sort of this self-denying prophecy. I got all sorts of people upset about various accusations I made, and they fixed the problems, to some degree. Now a lot of the fragility is still there, the spam problem is still out of control, viruses are still a problem. We grad students, when we were building the Internet, did not put economics in and we didn't put security in. It's still not in, and there continue to be problems. There are outages all the time and no one's keeping very much track of them. So I claim I was sort of right. But 1996 came and went, and I accepted an invitation to speak at the Worldwide Internet -- some conference whose title was sort of like the Worldwide Internet Conference in Santa Clara, California -- wherein I was expected to explain my failure to predict the collapse of the Internet. What I did is I carefully arranged the most successful publicity stunt of my career as a columnist. I planned to eat my column. By the way, I had promised that I would eat my column if this prediction had not come true. I accepted this speaking gig with the intent of making good on my promise and eating my column. I'm so proud of what I did. First of all, I rehearsed it. Eating paper is hard. In fact, you can't do it. Not only that, it could be dangerous. Some paper is printed with heavy metals and you don't want to be eating it. *InfoWorld*, fortunately, is published with soybean-based inks with no heavy metals. I got that all verified. The night before, I actually at the column and I realized you can't. Try it. You can't. It's really horrible, and it can kill you. I got a blender, and some water, and a bowl and a spoon, and they were in the podium when I came out on stage. So I come out on stage. There was a thousand of these Internet people, my peeps, the Internet intelligentsia, who are generally hostile to me by this time, because of all my gas bag ranting. So I began speaking and the audience was rumbling. I said, "You know I made this prediction in '95 that the Internet would collapse last year, and it did, just like I said." "Boo!" They started booing, because they could see me weaseling my way out of my commitment. I said, "Well all right, I was close. I was within a factor of eight. There was a 118 megalapse instead of a gigalapse." It was close enough. I mean it was within a factor of ten. Surely, that should be enough. Audience growling, rumbling, shouts of derision coming. I said, "Okay, okay. I'll eat my column." I had them roll in a huge cake that looked like my column. With icing, it had been made to look exactly like my column. I went down and I started cutting cake and eating it and handing it out to people as if this would suffice. Then I went back to the podium and I'm eating the cake. The audience was not going to buy this, so they started yelling. Finally, I said, "All right, okay, all right." I got an actual copy of InfoWorld and I tore out the offending column and I went through the audience, "Is this the column right here? Is this the one I promised to eat?" I got them all to agree that it was and I took it up on the stage. Then I started tearing it up into little pieces, and then I reached down and I took the blender out. Of course, when I took the blender out, then they all realized that I'd been planning this all along. I took the bowl out and I put the water in the blender, and I dumped the column in the blender and I blended it for a while. Then I poured it into the bowl. I asked them, "When you have soup, do you drink soup or do you eat soup?" They had to agree that you eat soup. So I was about to eat my column. You see, I didn't want to have a technicality. I didn't want to just drink it, because drinking is not the same as eating. I got them to agree that you eat soup, and that this was a soup and it was my column, and then I proceeded to eat the whole thing. Then I held the bowl upside down over my head, the way you do with an empty beer stein, to prove that I had eaten the whole damned thing. Anyway, this made the cover of *Barron's Magazine*. It made articles in *Wall Street Journal.* Every big magazine picked this up. This was such a successful journalistic pundit promotional event that I became better known for having falsely predicted the collapse of the Internet and thereby eating my column than for inventing Ethernet. That's how successful that PR stunt was. But generally [in] the community, the net reaction was positive. The fact that I actually ate the column was palliative. That is, people said, "Well no one ever does that. No one ever admits they were wrong. No one ever actually does that. Metcalfe did. He said he was wrong. He ate the column right in front of us. We were all there." Not only that, not only did I manage to generate all this interest and all this, I think, preventative worry about what was broken about the Internet, the self-denying prophesy. In the end, I ended up getting a positive boost [more] than a negative one on this. I'm reading all the time people referring to my ill-fated prediction.

Shustek: I trust you had no ill effects from the meal?

Metcalfe: No. Remember, I ate the column twice. I ate it the night before, and then the day of, and no ill effects, no problem.

Shustek: To the extent that it was a self-denying prophesy and that you had a beneficial effect on fixing the Internet, do you think the same kind of mechanism occurred a few years later in the Y2K issue?

Metcalfe: Yep, absolutely, all that hype. I was a journalist at the time, a publisher even. That was great for us, because it created controversy and it was self-denying. People were preparing for it long in advance, and it didn't happen. A self-denying prophecy.

Shustek: You've been critical of Microsoft as a monopolist, at various times. But one of the things you once predicted was that when Windows 2000 gets here, proliferates, that Linux would disappear. That hasn't happened.

Metcalfe: I didn't exactly predict that, either.

Shustek: Okay, what did you say?

Metcalfe: I'm sorry, do you have the exact quote there? Do you want to read it to me?

Shustek: I think it said, "When Windows 2000 gets here, goodbye Linux."

Metcalfe: Well, that was clearly wrong. <laughs> “Goodbye, Linux.” Well actually, there are other predictions I’ve made. I had to make a prediction a week, so I had eight times 50 predictions.

Shustek: You become a target.

Metcalfe: Yeah. I think my batting average is pretty good, but that’s one, that particular sentence, I would have to regret. On the other hand, if you were reading the Linux media at that time, Windows was doomed. I’m sorry, but Windows has not… Have you noticed Windows sort of dropping off since I wrote that? No. And they were also falsely counting, because Linux people generally didn’t care about security. When you went to count Linux nodes in the network, there they all were. But Windows people are more serious people. They have businesses to run, stuff like that. They’re not just grad students. I like to be nasty to open source Linux people. They were undercounting the Windows machines and overcounting the Linux machines and claiming that Linux was taking over the world. Linux has been doing fine, but it didn’t kill Windows. Vista just shipped today, and this is going to do just fine. I’m negative on both Linux and Windows, frankly, because I believe they’re 25 years old and they need serious fixing.

Shustek: Do you see any opportunities for something to replace them?

Metcalfe: I’ve written that I’m hoping that such a thing would occur. Maybe it’s going to happen through the evolution of Linux and the evolution of Vista. But if someone’s going to come in completely new, it’ll be on cell phones. That’s happening now. There’s five, ten operating systems on cell phones now, and most of them are not Windows or Linux. There’s some Linux and some Windows, but there’s some not, so that would be the next opportunity. We’re also pursuing the opportunity in the embedded space. There are ten billion microcontrollers shipped every year, and there’s no Windows for microcontrollers. There’s ten different raggedy embedded operating systems out there. So the opportunities for new OS’s are probably in cell phones and embedded more than PC’s, because PC’s are kind of passé now. They’re clunkers.

Shustek: Maybe a new operating system can evolve in some other ecosystem and then expand out to the desktop.

Metcalfe: It’s possible. OS’s are too familiar to a lot of the software I’ve dealt with in my technical prime, and that was in the ‘60’s and ‘70’s and ‘80’s. Why is that junk still around? Why are we still telnetting? Why do we still have command lines? Windows and mice -- we had those at PARC in the ‘70’s. What’s that stuff still doing around? Not only that, whenever a computer in this house goes awry, my wife expects me to fix it and I can’t. They’re kludges. They’re horrible, fragile things. And they’re good enough for a billion people.

Shustek: What have we done wrong in the industry that lets that happen?

Metcalfe: Not that we did anything wrong. It's just that the status quo in general, in this whole process of technological innovation -- which is what I believe my career is, technological innovation -- the status quo is strong and has lots of defenses. It's hard to innovate. There are barriers to entry, to new innovations and they're higher than they should be. Just to go back to your mentioning of Microsoft, I believe I wrote the first column attacking Microsoft for anti-competitive behavior. I did it in *ComputerWorld,* and I did it in 1991. It was after my bad experience, direct experience with Microsoft at 3Com where I witnessed face-up what it feels like to be on the receiving end. Now I had previous experience with AT&T, but this was now Microsoft. I didn't exactly criticize them for being monopolies. I exactly criticized them for anti-competitive behavior, which is different, because I think Microsoft has earned its monopoly. Where it went wrong is when it began to use its monopoly to deter competition and be a barrier to entry to new innovations. For example, the famous example is when they introduced [Internet] Explorer. They didn't have to use their monopoly on operating systems to kill Netscape. They could have killed Netscape even-steven, fair and square, but instead, they bundled. They bundled Explorer with Windows, which is a violation of law. I mean the Sherman, Clayton, one of those, and case law, says you're not allowed to bundle. You're not allowed to take a monopoly position in one field and then bundle a non-monopoly product, and thereby become a monopoly there. Someone realized a long time ago that that deters innovation and prevents new starts from innovative companies from coming along, so that's made illegal. What Microsoft did wrong was not to succeed, -- and I think Bill Gates is a fine man -- it's, I've referred to them as the Hitler youth -- that was, perhaps, strong language -- but the younger people at Microsoft, believing, sort of trying to emulate Bill Gates, but not really getting it quite right. Gates is an ethical, hardworking, shrewd son-of-a-bitch. But the Hitler youth, as I call them, they thought they were emulating Bill Gates when they started acting anti-competitively. They didn't have to bundle Explorer. They could have attacked Netscape fair and square with a little preparation to be sure that that competition could occur on the Windows platform. They could have done it that way. Instead, they chose to do it in an anti-competitive way. The thing that prompted my column in *ComputerWorld* is I attended a press conference for the introduction of a pen-based operating system. What was it called? GO, or some predecessor of GO.

Shustek: EO

Metcalfe: EO, some predecessor of EO, the one just before EO. Anyway, it was a pen-based OS. In San Francisco, I was a journalist. I went to the press conference. The day before, there had been a press conference by Microsoft to introduce Pen Windows, and it took all the air out of this press conference to introduce this pen-based operating system. The thing that was true is there was no Pen Windows. Do you use pen Windows a lot? No. There never was Pen Windows. Microsoft saw that this little company was going to introduce a new operating system, and scheduled a press conference the day before. That is anti-competitive behavior. Previously, IBM had been sued on this. They introduced the 360/90 in order to kill the CDC 6600. They didn't have whatever machine it was, the 90, or the 95. The justice department said, no, you're not allowed to introduce fake products in order to disrupt the introductions of your competitors. No bad monopoly. You're not allowed to do that. Well, here's Microsoft doing it. They did it to this pen OS. That's what provoked my column. So just to be clear, I have been and still do attack Microsoft, not for succeeding, but for turning its success into anti-competitive behavior, thereby slowing down the innovation process by deterring new starts from starting up. It's okay to serve customers better; it's not okay to damage competition, if you see the distinction.

Shustek: Do you see that anything has changed in the intervening 15 years? Do you predict anything might change now that Gates is less involved?

Metcalfe: Well first of all, Microsoft has been convicted of anti-competitive behavior, and that conviction is having its effect. That is, they're being sued, still, and they're having to behave themselves a little better, as happened to IBM during its heyday. It was forced to clean up its act a little because of this legal action. Balmer and Gates are of the same vintage. Balmer's an excellent guy, Gates is an excellent guy. I don't think the company is going to somehow get mediocre, suddenly, because Gates is doing his philanthropic thing. What was true of IBM in the '80's is that it was generally run by mediocre people after years of monopoly. That's not true of Microsoft. These days, Microsoft is being run by and is attracting some of the best people in the world. It hasn't yet gone down the tubes. I once predicted that Microsoft was going down the tubes. I hastened to add it was going to be a long tube. <laughs> See, I laugh at my own jokes. My wife complains about that. <laughs>

Shustek: Other companies like Google are also hiring excellent people.

Metcalfe: Well, Google is…. Microsoft's not going to last forever as the dominant player. Just look at the Roman Empire. Five hundred years, that's sort of the upper limit. <laughs> Maybe Google's the one, but then right behind, what's going to happen? Is it going to be Google forever? No. Google will somehow… But the timescales of these evolutions are way beyond the capability of covering in a daily newspaper or a weekly magazine. It would take decades, so it's hard to be right about them.

Shustek: After a decade of being a journalist, you decided to move on to another phase of your career and become a venture capitalist. Describe why it is you did that, and why you thought you could do a good job at it.

Metcalfe: I have a short attention span. It's about ten years, roughly. After about ten years, I sort of want to do something new. That's what happened. I did journalism and publishing for ten years. I was doing a column on this company -- it's in a funny place; they gave me directions. I like to write about new technology. One of the things I liked to do for my readers was to say, "This is going to be really important." My idea of a scoop was to identify a technology before other people had written about it, so that's why I ended up writing about a lot of startups, because that's where you'd find such a thing. I once heard about this startup called Ucentric in Maynard, Massachusetts. They gave me the directions: out route 2, left, right, left, right, and I realized they were directions to the [woolen] mill, the old DEC mill. They should have just said come to the DEC mill, but they didn't know that that shorthand was available. Here was the mill, packed with cars like it had been in the old days of DEC. Only there was no more DEC. It was just 100 startups had infiltrated the mill. Here I am interviewing this company, Ucentric, and there was a venture capitalist, Mike Hirshland, sitting there, which is kind of odd. I'm interviewing a company and their VC is there. I spent most of the day there. At the end of it Hirshland invited me to come visit him over at his VC firm, Polaris. The next thing I know, I'm a venture partner there. They invited me. What I viewed that was as a transition of career. Technological innovation is still my business, but this is a new role. Instead of helping people to better buy information technology, I was going to go start some companies, dealing with entrepreneurs, who were my favorite people. Scientists, engineers and entrepreneurs are my favorite people, roughly speaking. There's a four-way deal there.

The scientists develop the science. The engineers, they make it into products. The entrepreneurs organize the companies, and the VC's fund them. That's part of the technological innovation process. I viewed it as a way to keep in that business, but from a different point of view. Just as I had learned how to sell advertising, I was now going to learn how to be a venture capitalist, which is a completely different trade craft involved there.

Shustek: Did you think that your experience having started your own company would help you evaluate and mentor these startups, and has that been true?

Metcalfe: One of the conceits of my practice as a VC is I think that. I think that having built a company, I'm in a better position to choose and help companies. But along those lines, I think in the last six years, I've learned that I need to be putting more time in choosing than helping. There's only a certain amount of helping you can do, and if you've made a bad choice to begin with, it's very hard to help a bad choice into success. It's much better to put the energy into selection than helping. One of the things I had done in my early years as a VC is be chairman of the board, which is something VC's don't do a lot of. I did more of it because I thought I was more qualified, because I had been a chairman, an actual chairman of the board. I'm doing that less now. I'm retreating from that role, because it was a conceit that I could be so helpful to these companies. As an investor, I need to pay more attention. I'm going to keep helping, of course, but that I need to put more emphasis on choosing than helping.

Shustek: Do you find it frustrating that in your role, you provide the funding and you provide advice, but they sometimes don't listen to your advice?

Metcalfe: They don't? Sometimes they don't listen to my advice? How about sometimes they do listen to my advice, rarely. <laugh> No, see, I had ten years as a journalist where I was giving advice to a million people and they weren't listening. Now I give advice to a much smaller group and they're not listening. It's pretty much the same thing. I try to be helpful. One of my sources of compensation is to be acknowledged for having [helped]. "Oh, that was a great thing," or "Thanks for that help. " I do try to help the companies. But you're right. I have a portfolio now. I have eight companies. Well, we have a-hundred-and-some companies that all of us look for. I have eight that I sort of pay careful attention to. I have to be careful that in each of those eight cases, the people I'm dealing with there, they have one company that they're interested in. They're one-eighth of my portfolio, and that requires a different mindset. I mean, they're in charge. I can advise. I can invest or not invest, or advise, but ultimately, it's those teams that make or break the companies. By the way, I've learned in six years that the shortage, the scarce commodity in my practice, is not money. There's plenty of venture capital. Don't let anyone tell you there's not enough venture capital around. There's billions of it, gobs of it. And there's no shortage of ideas. MIT alone generates enough ideas. You could start a company a week. The shortage is CEO's, those people who can start those companies and grow them. If we had more CEO's, we could fund more companies and there would be more innovation.

Shustek: You said that one of the critical things is to select well. Do I take it then that the selection of the team is even more important than the selection of the idea?

Metcalfe: Well, yeah. This is a perpetual debate in venture capital: is it the market, is it the people or is it the technology? Which is it? Of course, you have to have all three. I guess what I just said was tantamount to saying that the people are the most important of those three. I guess, yeah, you would argue that a good team stands a better chance of making the most of the situation. On the other hand if the team is in a losing situation, there's not much they can do. You find a lot of mediocre people who just happen to be in the right market and they look like geniuses.

Shustek: Sometimes a good team can change the situation, when they see it's not working, into one that will work.

Metcalfe: That's what I tell our teams. Occasionally, the team will say, "Our market is not really taking off." And I say, "Aren't you a leader in the market?" "Yeah, we're the leader in the market." "So the fact that the market isn't taking off is your fault. Make the market take off. What do you have to do? Don't tell me the market's not taking off. Make it take off!" That's to your point. But there's luck involved. You see some mediocre teams who stumble into these huge vortexes and they all look like geniuses. You don't really know they're not geniuses till later. Then there are the really good teams that, owing to the ebbs and flows of luck, end up in a losing situation and there's nothing they can do. Venture capital is a very specialized, very subtle trade. I'm still learning it after six years.

Shustek: What sort of companies are you responsible for at Polaris? Are they mostly communications and networking, or have you branched out?

Metcalfe: A key reason I'm at Polaris is Polaris is diversified. That's our number one strategy. We are not a "dot com" VC firm, or even a telecom firm, or even an IT firm. We're diversified across technologies, including life sciences, drug delivery, medical devices, enterprise software, networking, networking hardware, networking software. It goes on. We're diversified. But still, my companies tend to be closer to what I know about. There is a notable exception, which I'd like to get to. I'm chairman of a company called Ember that networks embedded microcontrollers, of which, as I mentioned, ten billion are shipped every year. That's sort of a recapitulation of 3Com: networking a layer of computers. PCs were new, 3Com went to network PCs, and we succeeded and the rest is history. Now embedded computers are coming, so we're developing CMOS radios and protocols appropriate for 10 billion embedded microcontrollers.

Shustek: It's not a new field; people like Echelon have been trying to do this for a long time.

Metcalfe: They have. It's not a new field, but it's still a new field. Echelon is a fine company but they're not doing the job. There are still 10 billion un-networked microcontrollers out there. The pitiful fraction that they've managed to network-- they are hung up on wiring. You're not going to network those 10 billion microcontrollers with wires, I'm sorry. They keep writing articles about how wireless doesn't work. It reminds me of columns I wrote in the '90s when wireless didn't work, in the early '90s. I wrote columns; actually I went a little overboard: I said wireless will never work, which is going too far. But now I'm on the other side. The problem with Echelon, a fine company -- in fact 3Com almost merged with Echelon, and Ken Oshman's a fine man, and Mike Markkula is a genius -- but they're stuck on wiring. They'd better get off wiring or little Ember and a bunch of other companies are going to pass them by. Because those 10 billion embedded controllers need to be wirelessly networked. It's clear. The cables are too expensive for one thing. So echelon better get with it.

Shustek: What other companies are in your portfolio are interesting?

Metcalfe: They're all interesting, all eight of them. I've just recently stepped down as chairman, but I'm still a director, this is part of my altering role, I've recruited a great chairman for SiCortex, maker of supercomputers. Open source, Linux clusters.

Shustek: Open source? You haven't been a fan of open source.

Metcalfe: Open source. I'm flexible. It's much more important to be right than to be consistent. SiCortex says there's an emergent -- in supercomputing, in technical, scientific supercomputing, not enterprise computing -- there's an emergent platform. It's called Linux MPI open source. The reason open source is a special attraction to these folks, these are people who own their own codes. They don't buy software, they write their own. It's very technical. They don't buy Oracle databases, they write their own codes. So recompiling the OS is something that they really love to do, several times a day probably. SiCortex has noticed the emergence of this Linux MPI platform into the top 500 supercomputers in the world and decided, well, they're cobbling these clusters together out of old PCs using Ethernet as a fabric. So let's preserve the platform but put purpose-built hardware underneath it. We'll execute MPI over Linux and run Linux better, faster, cheaper. Improving delivered performance by factors of 10 per dollar per foot per watt. We start out with a 64-bit floating point microprocessor that has less than a watt of power consumption instead of more than 100 watts of power consumption. There's a factor of 100 there that we're starting with. Then we have a fabric which replaces Ethernet in the cabinet, and it's integrated on the chip. The core of this company is a chip. Then we put the chips on boards, and we put the boards on boxes, and then we sell the boxes. Our mission is to deliver teraflops for milliwatts. That's really cool. The chip just came back from TSMC a few weeks ago and is now running Linux and sending packets. I got an email from one of the guys running Linux on this chip. We will be shipping computers this year, and that's exciting.

Shustek: It's an exciting field, but of course it's a field that's littered with corpses of a lot of companies that didn't make it.

Metcalfe: Right, and the people who work at SiCortex were at all those companies that didn't make it, and they learned three lessons. See, three? Isn't that recurring, three lessons? Three mistakes we're not going to make: We're not designing our own microprocessor; we're not developing our own operating systems and tools; and we are not asking our customers to reprogram their applications. We plan to do this company on $50 million instead of $500 million.

Shustek: You mentioned that one of your eight companies was not in the computing communications field. What is that?

Metcalfe: Oh, well, yes, I've noticed that one of the big problems in the world today is energy. The world needs cheap and clean energy. Cheap and clean. Not just cheap, not just clean, but cheap and clean energy. The world needs it.

Shustek: Other people have noticed this too.

Metcalfe: A lot of people have noticed it. But I've noticed it, and I've noticed that too many of the people who have noticed it are Luddites and Greens and Marxists and politicians and lawyers and people who are in no position to solve the problem. However, scientists, engineers, entrepreneurs and venture capitalists -- we can solve the problem. So just like it took us 30 years to break the back of the communication monopolies and build the internet, we are going to take the next 30 years to break the back of the energy monopolies and meet the world's needs for clean and cheap energy. Cheap and clean. Cheap and clean. Both. Has to be both. Can't be just cheap, can't be just clean. It's got to be cheap and clean. This investment, my first energy -- I've sort of made two -- the first one is called Green Fuel. It's an MIT spin-off, which is one of my specialties, hanging around MIT and looking for opportunities. Listen to what Green Fuel does: It takes the flue gases from power plants -- oil, gas, coal -- and bubbles the flue gases through a slurry of algae, in the sun. The algae eat all of the CO2 and all of the NOX from the flue gases, so that the resulting effluent is free of greenhouse gases. By eating this CO2 and NO2, the algae double in mass every few hours so that once a day you're able to harvest them. From the lipids you produce bio-diesel, and from the starches you produce ethanol, and from the proteins you produce feed. Thereby killing two birds with one stone, solving -- not solving, but contributing to -- the reduction of CO2 emissions into the atmosphere, thereby the alleged impact on global warning. And second of all, producing this huge revenue stream of bio-fuels. I've just been to the desert of Arizona watching them build their third generation, slightly scaled up. It's now about a third of an acre. These are high tech greenhouses, basically, that take a lot of acreage, ultimately thousands of acres next to the power plants to recycle their CO2 into bio-fuels. Isn't that cool?

Shustek: It depends on the existence of these old technology power plants.

Metcalfe: Well, you think those power plants are going to go away tomorrow? They're planning to build hundreds of gigawatt coal plants in the United States and China. Coal is the cheapest fossil fuel available. It's also the dirtiest. Green Fuel is a way to use coal without putting the CO2 in the atmosphere. I'm sorry if you're thinking that… I'd love for us to build a thousand nuclear power plants in the United States as soon as possible, and that's a whole separate argument. But that's a whole separate argument. In the meantime Green Fuel is going to make it cheap and clean. Cheap because of the recycling. CO2 is not a pollutant, it's a valuable plant food and we're recycling it. So there's great value generated. And clean. Cheap and clean is the Green Fuel proposition. Let's just hope they [knocks on wood] succeed.

Shustek: Good luck with it. You said you had another company that was an energy investment?

Metcalfe: Yeah, that's on the other end. That's the generation side. There's generation, transmission, storage and consumption. On the consumption side, we've invested in a battery company. I'm sorry, on the storage side. Well, it's actually on the consumption side, too, when you think about it. These are micro-batteries. There are a lot of battery companies, you know, electric cars and other worthwhile activities. This is a micro-battery company that makes batteries as thin as a postage stamp. They're indestructible, infinitely rechargeable, thin film lithium ion, solid state batteries used in embedded applications – embedded, again -- to power those 10 billion microcontrollers. Those would be used in energy management, security and a million other applications. Convenience, smart cards, cell phones and so forth. Not the main battery in the cell phone, the back up battery in the cell phone; the real time clock battery. We have a solution for that. We hope to ship billions of those soon.

Shustek: For those eight companies, you serve as the primary VC liaison?

Metcalfe: From Polaris to them. There are other VCs involved in all of them.

Shustek: So you work as a team in Polaris? You're not an individual investor?

Metcalfe: No, there are 14 of us, we meet weekly, we consult on all the investments. There is specialization. I have these eight companies that I'm responsible for, but it's highly consultive with the team. Then at the company there's other VCs. The board is usually a syndicate of two, three, four or five investors.

Shustek: Are you enjoying it? Do you think you'll fill out your decade-long attention span being a VC?

Metcalfe: Oh, yes. I want to be the best venture capitalist that ever was. I want to be better than Vinod [Khosla]. And John [Doerr]. And Don [Valentine].

Shustek: And Arthur [Rock].

Metcalfe: And Arthur! Let's not stop. Let's go to Arthur, and all those guys. I want to be better than them, or at least as good, because there's fun in mastering something and being good at it. I haven't been at it long enough to really… I'm not expert enough. I refuse to give talks on how to be a good venture capitalist, because I don't know yet. But I'll start giving those talks in a couple of years, I figure. I haven't begun thinking about what my career will be after that. But I will in three or four years.

Shustek: What other activities are you involved in? I know that you have been or are still on the MIT board?

Metcalfe: Yes, I'm a life trustee of MIT, and it's enormously important and fun work. As a result of that affiliation, I'm on the board of Technology Review magazine; I'm the chairman of the Leadership Board of the McGovern Institute for Brain Research at MIT; I'm on the visiting committee of the electrical engineering and computer science department, and I'm active in many entrepreneurial activities: the

100K competition, the idea stream conferences. MIT has an elaborate infrastructure for supporting and encouraging entrepreneurship.

Shustek: Are you teaching these days?

Metcalfe: I give an occasional lecture; I don't have any responsibilities. Today, for example, I'm giving a guest talk at a seminar at Boston University up the street here, in the history department. I sometimes talk at MIT. I gave a recent talk on collective intelligence, as if I knew something about that. [It's] related to Metcalfe's law; see, Metcalfe's Law and collective intelligence, you can see how they might be related. I give a talk, repeatedly, once every year or two, on writing: the importance of writing and engineering. I don't talk to the people who write at MIT, I talk to the engineers who don't, convincing them [that] to be an engineer you have to know how to write and speak. They think it's like a choice: I can either be an engineer or I can write and talk. No, no, you don't understand; you can't be a good engineer without writing and talking. So I give that speech now and then. And I often, too often, actually, I like speaking and I get invited a lot, so I do too much of it.

Shustek: Along the lines of your advice to engineers to learn how to write, what other career advice would you give to people who want to be involved in high technology? In terms of what sort of education to get, what sort of first job to get?

Metcalfe: Well, the first advice to give is that you should be in science and technology. That is, technological innovation is the source of all progress, so you should be in the technological innovation business, at the core of which is science and engineering; science, engineering management, I would say. So you should be in that. It's the highest calling, to be in technological innovation. Democracy, freedom, prosperity -- they all stem from technological innovation. All of it. I'm sorry. If you disagree with me, fine. Freedom of speech guarantees your right to say so, but I disagree with you. You should be in science and technology and math. That's where to be, that's where the action is, and the world needs more of us. Second, you do need to learn how to communicate. In addition to your technological, scientific, managerial expertise, part and parcel of that is knowing how to communicate: how to speak, how to write. I like to give a lecture on the importance of being the person they choose to stand at the whiteboard. How that is a control position, and an important position in the evolution of ideas. Being the person they say, "Bob, would you stand at the whiteboard and write down what we're saying?" That's a job you want. You have to have good penmanship, but then you have to respect the control you have, because what you write down is a paraphrase. What you heard and how you choose to paraphrase it is all-important in the evolution of that conversation. That's one of the control points in moving things forward. Being good at that is important, both to the team you're on, but also your own advancement. Then I like to write about and give speeches about selling. The typical tragic flaw of scientists, engineers and entrepreneurs, especially technologically oriented ones, is that they think sales people suck. "Sales people are stupid, and fat, and empty-suits; they're lower than whales," or whatever. That is a tragic misconception. Sales people are a different form of life, I will grant you that, but they are every bit as important and skilled as engineers and scientists. If you try to start a company and not have any sales people, that's the end of your company, generally speaking. I give that pitch a lot. That nothing happens until something gets sold. So you'd better learn to sell. Sell your ideas as an engineer or a scientist. How do you think people get Nobel prizes? Do you think they just sit in their labs and people all say, "This person should get a Nobel prize." No, when you look behind these Nobel prizes, there's a sales campaign out there, organized by the university or the contributor. If you really want to win a Nobel prize you've got to learn how to sell, or you're not going to get the Nobel prize. That's just the way it is; I'm sorry. So selling's important; I give that pitch. Then the special nature of small companies versus large companies. I have a whole spiel about that. For example, in a small company, the company grows faster than the person whereas in the big company it's the other way around. What does that mean? What do you do differently? Well, recruiting's important. You don't "hire" people. This is a pet peeve of mine. They say, "We're going to 'hire' a CEO." I say, well, I'm sorry, you don't hire good CEOs. "Hire" sort of connotes there is a long line of people applying for the job and you're going to interview them and choose which one. Well, it's not like that. If you want a good CEO, you've got to go out and recruit them, you don't hire them. That's part of my spiel on the difference of small companies versus big companies.

Shustek: Do you recommend to people at the early stages of their career, that they get some experience in large companies before getting into small companies?

Metcalfe: I do. Well, I say you should complete your education and do graduate work at a big company. Then the smart alecks say, "What about Michael Dell, Steve Jobs, Bill Gates. What about them? They dropped out of college and they never worked for a big company." And I say, those are what we call exceptions to the rule. As a venture capitalist I have a moral imperative: I never induce anyone to leave school to start a company. It's just a rule I have. I consider it immoral to do that. For example, in the case of Ember, the founder there, Rob Poor, was a grad student when I met him and he said, "I want to start a company." I said, "Well, I'll talk to you as soon as you get your Ph.D." Which he then did. I feel good about that. I could have said, "Oh really, you want to start a company? What would it be like, have you written a business plan, would you like to come over and have lunch with me, dah dah dah", and start the whole process. I wouldn't allow myself to do it. Complete your education, go as far as you can as fast as you can in your education, and go to graduate school at a big company and learn how companies work. Then you're qualified to start your company.

Shustek: Would you recommend people to concentrate on computer technology, or do you think that it's so far along in its evolution that bio-medicine is the hot area for the next few years?

Metcalfe: There are many ways to live, and so you have to be careful. For example, I don't mean to say that being an entrepreneur or a technological innovator is the only viable lifestyle. Similarly, to say that "bio" has eclipsed "info" as the place to be, I sort of feel that's true, but that doesn't mean that there's not a lot to be done in info, and bio's going nowhere without info, because now bio is info. But I have to admit that were I entering graduate school now, I would be in something bio. And if not bio, something chemo. And if not something chemo, something material-o, you know, nano. I admit that. But there's a lot of info, in fact, all of those fields are largely now info, so it's really hard to be in any of those without also being info. My favorite example of that is Eric Lander, MIT, the guy who played a key role in sequencing the genome. He's an information technologist who used to be at the Harvard Business School, and here he is sequencing the genome. How did that happen? It's because info and bio are almost inseparable now, and that's true of almost everything. I guess it's still a good idea to know info. I'm trying to get a teacher chair endowed at my daughter and son's prep school out here. They're a snooty New England prep school, and they're interested in the liberal arts. The trustees are resisting me because they view computer programming as a trade craft. I'm making the argument that computer programming is a liberal art now. That you need to have some base level of literacy in, literally, computer programming. That's a liberal art now. The definition of a liberal art was not frozen in 1492, you know, it's evolved, and computer programming is now a liberal art. It should be taught at this little prep school, so we're going to endow a chair there and get a teacher and teach computer science to these kids.

END OF INTERVIEW