Interview of Vinton (Vint) Cerf

Interviewed by:
James L. Pelkey

Recorded: February 8, 1988
Reston, VA

CHM Reference number: X5671.2010

Vint Cerf: My name is Vint Cerf, and all the comments that I make today should be attributed only to me personally, and not to any corporations I've ever been associated with, nor with the US government, nor any other agency.

James Pelkey: Thank you very much for your time. I've given you some background. The area that is of most interest to me, although you can comment on any of them, is particularly the theme of technology and protocols and applications and, from the perspective of how did the intellectual property that was the basis for the communication industry, how did it come about and how did it evolve?

Cerf: Ok, I can't speak to the telecommunications industry as it developed in the point-to-point mode. I don't know anything about the history of the development of modems and things like that, so my experience base is at the point where packet switching was introduced. That was back in 1968, the Advanced Research Projects Agency, now called the Defense Advance Research Projects Agency, had a problem. Namely, it was buying computers for various research locations around the country, like MIT and Carnegie-Mellon and Stanford, and each year, each university would ask for another, higher speed, more up to date computer. Eventually, ARPA couldn't afford to purchase computer equipment for every single university that needed it to carry out its computer science research work, so the question was: how can we make it possible for ARPA to invest in this equipment, but make it available to a much broader range of people than just the ones at the physical location where that computer was sited? The answer was we needed some kind of remote communications capability. Now, bear in mind, in 1968, the idea behind timesharing was only just beginning to emerge. It was something that John McCarthy at Stanford had thought about when he was at MIT, and brought the idea about sharing of a mainframe, Digital Equipment Corporation’s PDP-1, to Stanford. So, it was in the context of ideas about timesharing, the Multics project at MIT, Project MAC which was getting started, those things combined together with the question of how do I get remote access to a computer? When ARPA went to talk to AT&T, which at that time was the place to go if you were asking about telecommunications, the recommendation that they got back was just have a whole bunch of point to point lines connecting all the computers with all the sites that want to use them, till you get an 'N squared over two' grid of wires. It was too expensive. It was more expensive than ARPA could afford, even though ARPA was, relatively speaking, a well to do research funding organization.

Pelkey: Were you at ARPA at this point?

Cerf: No, at this point, I was at UCLA as a graduate student still. I had come there in '67 from IBM to go and learn more about how computers work. But I know the history of the Arpanet well, because I was associated with it from down in the technical end of things.

Pelkey: UCLA was one of the first nodes –

Cerf: UCLA was THE first node that came up, and I was responsible for the software of what was called the Network Measurement Center, to figure out what the performance was of that network, to try to understand how to drive it into bad behavior and things like that. In any case, the motivating factor for the development of something other than point to point or circuit switching, which was the other alternative to having lots of point-to-point wires, circuit switching took too long to set up. In other words, if you wanted to -- if you had a program running in a machine, and particularly if you were a user who was sitting there typing at it, and you wanted to interact with the timesharing system, you didn't want to have to make a telephone call, in effect, type something at it, get something back, and then shut down the phone call so you could go and talk to somebody else, and then switch back and call the other site again. It just didn't make any sense. It took too long. The switching speed was too slow. The alternative would have been to leave circuits up all the time, but then it's wasteful, because you weren't typing at it all the time, or it wasn't typing back at you. I realize I'm using the word typing here, you know what I mean. So, this burstiness drove everybody in the direction of finding some other alternative way of sharing broadband resources. And eventually the idea emerged, and I don't know whom to point to for the idea, but people like Larry Roberts and, well there's a whole collection of them. In fact, there is a history of the Arpanet that was prepared by Bolt, Beranek and Newman, which you can probably get a copy of from them. The person who can help you is Alex McKenzie at Bolt, Beranek and Newman Laboratories in Cambridge. He probably has a couple of copies left. In fact, I'm hoping to take that history and add to it for it only went up to about 1978 or so; sort of the first ten years of the Arpanet. I would like to put the next ten years on it and then publish that myself.

Pelkey: That would be a great story.

Cerf: In any case, there's a lot of material in there about the early history. What was important is that computers were becoming less and less expensive at the time. Costs really were coming down, and the first ideas behind –

Pelkey: This is '69 and '70.

Cerf: Well, it's still '68. The idea of using a small machine began to make sense, because minicomputers were just starting to come out. The Honeywell series, if you remember, the DDP-516s, 316s and 716s. I guess the first ones were the three and the fives, or maybe five came first and then threes then sevens. That may be the sequence. They were the first minis, really, out the door. Digital hadn't started selling PDP-11s yet. It hadn't happened, although I think PDP-8s were around. They were a little too small for this application. When ARPA said we want to build a packet switching system, they explained what it was supposed to do, the people from the traditional telecommunications world thought that it just wouldn't work. So, ARPA went out with the RFQ anyhow, and Bolt, Beranek and Newman won, and ultimately built the Arpanet. So the early history of packet switching was driven by an economic problem, namely, they couldn't find a way to pay for all the telecommunications circuits to leave dedicated lines in between machines operating at high speeds. People wanted high speed because they had file transfers to do. It wasn't strictly a matter of a person typing on another machine; it was machines sending data back and forth. That was my first introduction to the whole concept while I was at UCLA: receiving delivery of the first Arpanet IMP, followed by a visit from one Robert Kahn, who had designed the thing. The whole system was largely Bob's architecture, and he came out to UCLA to go and find out, by kicking the tires, how this thing would really perform. He had some theories about places where it would break that not everyone agreed with him about, but he said he was going to go and prove that it would break in certain ways by generating traffic at UCLA and forcing it through this four node network in various and sundry ways. So he and I worked together, me doing the traffic generation and measurement software and he was figuring out what experiments to perform for three or four weeks at UCLA, and that's a collaboration that has gone on ever since. So, the early –

Pelkey: Now, were the protocols at that point in time defined by BBN?

Cerf: Only the lowest level protocol was defined. There were several sets. First of all, inside the network there were lots of protocols required just to manage a distributed system. It had to do routing. It had to do flow control. It had to do interfacing to the mainframe computer. So they defined the internal protocols of the net, which weren't visible to the computers that attached and used them, and they defined the protocol for interfacing to the network. They defined the physical level of interface, the link level and the packet level interface. They called it BB&N-1822, because that happened to be the document number that specified the interface between the host and an IMP. So 1822 is something you will hear from time to time if you ever talk to these people, and all that was is the document number that specified what the interface was. The rest of the protocols, the ones that were in the host computers themselves, were left open to the R&D community, so there was something called the Network Working Group that was started and chaired by Steve Crocker. Steve was a colleague of mine at UCLA and took responsibility for trying to pull people together to organize the development and design of these various protocols. So it was Steve, who ultimately went to ARPA also from UCLA, who led all of that effort and developed the set of protocols, a host-to-host protocol, a protocol for letting terminals talk to host computers, regardless of what kind of terminal it was, sort of a device independent terminal oriented protocol. There were file transfer protocols, and there were electronic mail protocols. So, a lot of the things that we find commonplace and almost a necessity in the electronic world of communications today, at least in the R&D community, were developed by that collection of people. Of course over time, that collection grew to a large number. So Crocker led the first Network Working Group.

Pelkey: Who else, do you recall, was in that first Working Group?

Cerf: Let's see. I would be hard put to give you all the names. Darn. There were people like Jonathan Postel who was part of that group, Robert Braden, who was at UCLA in the computer center where they are very different -- it's still true. There are computer centers on campuses and there's, these crazy computer science places, and they are still segregated in many ways, but Braden was one of the few people in the computer center who cared a lot about and was interested in the R&D side, so he was involved in the early stages. Those are all UCLA people. A person up in SRI International who was very much involved in this project, particularly Douglas Engelbart, whose name ought to be familiar, maybe not. Doug started a project called the Augmentation of Human Intellect project at SRI International. That's another project funded by ARPA. For all practical purposes, it was the first hypertext experiment.

Pelkey: What date was that?

Cerf: 1966 or thereabouts, '67 at the latest. Engelbart's group invented the mouse, the thing that goes -- ok? He invented black on white displays instead of the conventional P40 phosphor. He had structured text. You could build a document that had a Roman Numeral structure to it -- had a plex of information, so everything was structured. I could go on for hours about what he called the On-Line System. His model was that computers ought to be an integral part of knowledge work, and that knowledge workers should have all of their work accessible by computer, so that the computer could file it away and make cross references. If you read an article generated in his system and you came to a reference, you could 'bug' the reference with a mouse and it would open that article up so you could read it.

Pelkey: Is he still at SRI?

Cerf: SRI sold that whole operation to Tymshare, so Engelbart is now a senior person in the R&D community at McDonnell Douglas, which as you may remember bought Tymshare.

Pelkey: And is Crocker still at ARPA?

Cerf: No, Crocker -- none of them ever stay in the same place. Crocker went from UCLA to ARPA, came back to UCLA to finish his PhD, went to USC Information Sciences Institute, and then to Aerospace Corporation to start a laboratory, and then joined Trusted Information Systems, which -- and runs it's west coast operation, and Trusted Information Systems is a special group headed by Steven Walker, who is ex-NSA and ex-ARPA and ex- Pentagon, where he was the Director of Information Systems in the Undersecretary of Defense for Research and Engineering's office.

Pelkey: So, you had this working group.

Cerf: We had a working group that mostly consisted of academics, and the problem they were confronted with is: what kind of software should we put in computers to let them communicate across a packet switching network? Can we make it really work, not just work the batch-like things, file transfer and electronic messaging, but also for interactive traffic?

Pelkey: Although as I understand, in the beginning, the IMPs were really meant to be host-to-host and terminals would be connected to hosts, and that –

Cerf: That is absolutely correct.

Pelkey: And the traffic started to get so large -- there was so much terminal traffic that they had to develop a new IMP that handled terminals connected directly to the net.

Cerf: That's true, but let me help a little bit here. It's true that the IMPs were originally developed only to interface to hosts in the belief that most of the terminals would be on the timeshared hosts locally. People had computers sitting in their labs and they had terminals hanging on them for timesharing purposes, and there was no need to have any other kind of IMP that could directly interface with a terminal because everybody would already be on a mainframe, and the mainframe would do all the protocols that were needed to allow the terminal to have access to someone else's resources remotely in a timesharing fashion. What actually happened, though, is that a lot of new users came along who didn't have mainframes at all. They had no timesharing system; they just had terminals and modems. The question was: how do I service them? Or alternatively, what if you were someone who had a mainframe, but you were traveling, and you didn't travel with your mainframe with you, back then. Today you've got laptops, but back then, it was, if you had anything, it might have been a TI-725 that weighed 27 pounds or 37 pounds. So the terminal IMP, or TIP was developed, based on a Honeywell 316, that could allow dial-up access, but it did all the protocols that the mainframes did to service the terminal. They didn't do electronic mail or file transfer, but just telnet. So, that's right, those came after the IMPs. The Network Working Group developed the idea of layering the protocols. This is not an OSI idea. It was something that had been -- the layering concept had been around certainly since the late 1960s and early '70s when this Arpanet Network Working Group was beginning.

Pelkey: Was that an invention or an intellectual property of that Working Group at that point in time, or did it reach back –

Cerf: It did not reach back, so far as I know, and we really believed that we were putting together the basic concept of layering the protocols to keep them separate from each other and having well defined interfaces so you could have many different protocols served by the next one underneath, and to have well defined interfaces so you could pull one out and put another one in.

Pelkey: Did that get documented -?

Cerf: It absolutely got documented. If you read any of the articles, of which there are probably now hundreds, you'll find the layering concept pretty well defined. In fact, there's another –

Pelkey: Are there references back to an original document done at that point in time?

Cerf: There are over a thousand internal documents that are openly available called Request For Comment, RFCs, and the RFCs have been and are still edited by Jon Postel. Some of these people have made a career of the experimental R&D data communications world, and Jon Postel is one of them. You should talk to Jon.

Pelkey: Where do I find –

Cerf: Jon Postel is at USC Information Sciences Institute. 213- 822-1511. He had a recent document that I will show you. I don't have all thousand of them here, but they are still coming out. This one is dated December 1987. It's RFC #1037. These are the documents that people have contributed as, literally: "This is a proposed protocol, or this is a proposed project. What reaction does the community have?" Eventually, some of these become accepted as working standards for this R&D community. God, there is so much to tell. This just -- some of this is just barely the beginning, because after the Arpanet got going, probably the biggest motivating factor to get all the protocols done was a demonstration of the Arpanet that Bob Kahn managed in the basement of the Washington Hilton Hotel. In November of 1972, there was the first International Conference on Computer Communication, and it was Bob's job, because Larry Roberts had promised to bring up a node of the Arpanet with its 50-kilobit lines going out to different parts of the country. By that time, '72 now, the net had been running for three years. There were 30 nodes on the net, and Bob orchestrated 50 different groups to put demonstrations on. People could sit down at terminals that were in the basement, connected to a TIP, the TIP was connected with two 56-kilobit lines to the rest of the network, and use the Arpanet. Not just hear people throw view graphs around about it, use it. They sat there and they couldn't believe it. It worked.

Pelkey: And who were some of the people who attended that conference?

Cerf: Bob Kahn would be -- he's the president of CNRI and you really ought to talk to him, and it's a shame that he's not here today. He's off at the National Academy of Sciences doing something, but Bob has all the data. In fact the most recent meeting of the TCP/IP inter-operability conference had a session on the 1972 demonstration, and he has films that were taken of the early Arpanet developers discussing how it all worked. People like Frank Heart at BBN, like MIT -- Licklider, J. C. R. Licklider who was a major force behind this thing, Bob Taylor who preceded Larry Roberts at ARPA; Bob has got a pile of that material.

Pelkey: If you would do me a favor and mention to Bob –

Cerf: Oh, I certainly will. He'll be interested to talk to you. So anyway, this demonstration of the Arpanet was probably the most critical turning point for packet switching as a technology, because it made it real for people including the ones in the telecommunications industry who didn't think you could do it.

Pelkey: Now, where were you at this point?

Cerf: At this point in '72, I had moved from UCLA to Stanford. I was on my way from UCLA to Stanford. The last thing I did when I was at UCLA was to participate in this demonstration to help -- I don't remember all the things I did, but I was involved in trying to make sure all the people were there and all the pieces got put together, and I did a few demonstrations of some of the network measurement tools that were in use at the time. And then I went to Stanford, at that point. So this was such a crucial turning point, because it demonstrated the reality of packet switching. At that ICCC, there were a number of people from around the world who were already beginning to get interested in packet switching, and even had done some work already. For example, in England, a guy named Donald Davies at the NPL (National Physical Laboratory) had built a single node packet switch for the laboratory. He never extended it to multiple switches, but his concepts were exactly similar to the way the Arpanet was working, but he did it independently. In France, a guy name Louis Pouzin and some of his colleagues, Hubert Zimmerman, and Gerard LeLann had started working on what they called Cyclades, which is a datagram packet switching network. Another guy named Remi Despres was responsible for the French Transpac development. Before that it was called RCP (Reseau de Commutation par Paquets), so that was their first -- that was the French PTTs experimental packet network. By this time, Telenet was beginning to be formed. In fact, Bob Kahn formally started Telenet at BBN. Remember, that BBN started Telenet. Bob Kahn and Steve Levy were the principal officers of that company, initially, until it was finally turned over for Larry Roberts to run when he left DARPA. Then it was sold, subsequently, to GTE for $60 million or something like that. So it really got started by BBN.

Pelkey: But Bob didn't get off –

Cerf: Bob didn't go -- no, no, Bob went, in fact in late '72, Telenet had gotten formed and Bob Kahn went to ARPA from BBN, so he started in ARPA in the end of '72, and didn't leave until 1985, late '85. I was leaving UCLA in '72 to go to Stanford to teach for four years, so I started at Stanford around November of '72. What happened after that was that the people who came together in November of '72 and observed that this thing really worked, and many of them were already interested in packet switching as a technology, said why don't we form some informal organization to figure out how we can interconnect these various packet nets that are being built. They asked me to chair that group. We called ourselves the International Network Working Group, INWG, and it finally got turned into a part of IFIP. It's now IFIP Working Group 6.1. And of course there's 6.2, 3, 4, and 5 now as well.

Pelkey: And IFIP stands for?

Cerf: International Federation of Information Processing, which is an international computer science body.

Pelkey: When did this work get turned over to what we now call CCITT?

Cerf: Wait, you need to understand that the prominence of the international standards did not directly stem from the R&D community. You need to realize how that actually happened. The basic concept behind packet switching was formulated in the Arpanet, and in some of the -- NPL networks, for example, and in the SITA, Société Internationale de Télécommunications Aéronautiques -- it's the air transport - - I got that all wrong. SITA is for air transport communications. They invented a packet net too about the same time. They may not have called it that, but that's what it was. All of that didn't quite wind up in the CCITT standards. The X-25 packet standard is something that really Barry Wessler and Larry Roberts should take credit for in the US. I'm not going to be able to generate the names for the other three countries. Remi Despres in France certainly gets credit for pushing X-25 through. In the UK it was another fellow, I'm sorry I can't pull his name out of my head, but either Larry or Barry could tell you, from the British PTT, and in Canada, it was Bell Canada who pushed very hard. At that point it may have been David Horton who was vice-president of Bell Canada, who was working with the Canadians on X-25. What happened is that after Larry had seen how useful packet switching was in the R&D world, of course Larry should speak for himself on this point, he went off to head Telenet, and then discovered that, from a practical point of view, it was hard to sell packet switching to somebody who didn't have any software in their mainframes to do packet interfacing. So the question was: how do I provide this service? His view was to start selling it by replacing high cost point-to-point circuits with switched circuits, virtual circuits, so it behaved as if it were an ordinary point-to-point circuit, but in fact, underneath, it was sharing the resources by using packet switching. So he designed and built, along with the other people at Telenet, a system that ultimately had a user interface that is now known as X-25. The Telenet interface wasn't quite X-25, but it was very close, and as Larry and Barry Wessler and Remi Despres and some of the other people in France and Canada got together at CCITT, or externally to it, to propose an international standard based on the virtual circuit concept, that's quite different from the way the Arpanet worked, which was datagram oriented. It was different from the way Louis Pouzin's work on the Cyclades functioned, and also from Donald Davies' NPL networks all those others were datagram oriented. So for years, there was this religious battle between people who had datagram style networks and people who had virtual circuit style nets. Marketing problems drives the virtual circuit guys; "how do I describe it to somebody?" And: "how do I bill for it?" They finally decided the easiest way to bill for it was to bill for having set up a circuit and for having sent a certain number of packets down that virtual circuit. Moreover, the software that was going to use it was accustomed to having either a switched or dedicated line, and it was easier to adapt that software to an X-25 virtual circuit style interface than it would have been to adapt it to a datagram style of operation. So that whole standardization effort in CCITT went down the X-25 virtual circuit path, while the R&D community generally stuck with a datagram style of operation, and you can see that in spades when you look at the Ethernets, because Ethernet is, if nothing else, a datagram mode of operation. It's everybody's out there shooting packets down the coaxial cable. What happened as a result is that the R&D community had to come to grips with a very much more challenging underlying communications environment where datagrams weren't guaranteed. If you sent one it might get clobbered by something else or it might get lost or it might get thrown away because of congestion, and the higher level protocols that operated on top of the basic datagram mode had to be far more robust and have more mechanism in them for flow control and retransmission and duplicate detection than the protocols that were evolving on the basis of virtual circuits. So these communities really went in different directions. They used to fight tooth and nail with each other, and I was out there fighting too. I was beating the table saying: "God damn it, it had to be datagrams because that was much more -- it required less of a network and you had to do things end to end anyway, because you wanted to have the mainframes assure the other end that they had really gotten the data, and not just that the network thinks that you got it," so there were a lot of arguments along those lines.

Pelkey: Let me interrupt. I'm confused on one point. The difference between virtual circuits and datagrams, both required either a switched line or a leased line.

Cerf: Yes.

Pelkey: So why –

Cerf: Why was there such a battle? Well, it was the switching mechanism that was at issue here. Either you talk to the packet switches and said: "Please set me up a virtual circuit and let me know when it's done and then and only then will I start sending data," and the other guy is saying: "I will hand you a piece of information with the address on it, and you do whatever you have to do to deliver it. Oh, and here's another one, but it's going someplace else." The arguments were on the application level. Sometimes you want an application that sprays data to 100,000 places, well alright, maybe 50 or 100, and I don't want to wait, I don't want to delay that application by having it wait to set up a circuit or, it takes resource to keep track of which circuits are set up and which ones aren't.

Pelkey: I'm not sure I understand, but please continue.

Cerf: Well, even if you weren't billing for it, just mechanically implementing a virtual circuit means that the source and the destination have to agree on the tag for the virtual circuit, what name is it, and that takes up table space. I'm not trying to make an argument today about any of this, because memory is so cheap, but in those days memory was expensive still and people argued about how to make most efficient the ability of a host to get rid of the data it needed to get rid of to talk to somebody else. So, to make matters even more complicated, a great deal of the R&D community who had started out in the packet world, was being supported by the Defense Department, and in the Defense Department's lexicon of networking, most of the concerns it had were related to highly hostile environments, where there was jamming going on. Parts of networks were being blown up. Some of it was mobile radio and you couldn't maintain a point-to-point sequence link anyway. People would go into radio shadow because you retransmit and you never hear them, so there were all kinds of reasons not to try to build virtual circuit type networks for the military, and that drove us all in a different direction. So, the set of protocols that evolved around the datagram world were more complex and substantially more robust . . .

Pelkey: It's kind of interesting, if I might jump ahead for a second, the datagram concept, the intellectual force behind that, really did go into local area networking.

Cerf: Absolutely, because local area nets were the direct analog of AlohaNet -- remember Norm Abramson and Frank Kuo, who took all the taxi radios and put some control on them, small microprocessors or even dedicated boards at that point, and were completely random carrier sense multiple access nets. Metcalfe, who'd been out in Honolulu for a sabbatical, came back to Xerox Palo Alto and said: "I ought to be able to do that on a coaxial cable, or a wire," and he did.

Pelkey: This is about '74, '75?

Cerf: Not even that late. It was '73, because by that time -- that must be right, because Metcalfe was thinking Ethernet by June or so of '73, I am reasonably sure, because I was at Stanford then, and he and I were meeting to discuss what protocols were going to be needed in that kind of environment, and about that time, Bob Kahn already had people working on packet radios, which were similarly sharing of a common -- carrier sense multiple access -- environment where you didn't have a nice clean circuit at all set up. There wasn't anything to set up. So, Metcalfe's Ethernet and Bob Kahn's packet radio network were all drivers in the datagram mode, and they pushed me, as the guy who was developing the higher-level host protocols, in the direction of something quite different from what had even been running on the Arpanet.

Pelkey: Could you talk about that a little bit more?

Cerf: On the Arpanet, there were a set of protocols that were based, in large measure, on the belief that the Arpanet wouldn't lose any data, and for the most part, it didn't, because if you looked way down deep inside the IMPs, you discovered they were setting up virtual circuits to handle the datagrams, so it really was a datagram network built on top of an invisible virtual circuit net. So they were really good about not losing data, but occasionally they would lose data, and when that happened, the old protocols broke. They just sort of stopped running, the Network Control Protocol, or NCP. It doesn't have anything to do with what IBM calls NCP, but it was the general term that --

Pelkey: And as I recall, in Arpanet, sometimes these nets would go down and have real problems as they were evolving these protocols.

Cerf: Certainly, the Arpanet itself had maybe a half dozen occurrences of real terrible disasters over its 20-year lifetime. Some of those are funny, in fact, and are worth an episode all by themselves. There isn't time to go through those right now, but I kept an anecdote file of all those things that happened. What had happened to us is that after we looked at the protocols we had developed for the Arpanet and asked ourselves: are they appropriate for a multiple network environment where many different nets were involved with different kinds of packet switching, broadcast satellite packet networks, mobile packet radio networks, Ethernets and Arpanets? The answer was: "No." None of those protocols were strong enough at the host-to-host level to cope with the kinds of packet loss that might be encountered in some of these nets, and if you traversed a particularly lousy net, or you traversed a net whose packet size was smaller than the net next door, so you had to break the packets up, how do you get them reassembled again? All those things drove us in the direction of a datagram oriented internet protocol. So Bob Kahn and I sat down and did some early design work and wrote a paper that got published in the IEEE in May of '74, called "A Protocol for Packet Network Interconnection," and that paper was the seminal paper behind the development of the TCP/IP protocols. It started out without recognizing the need for separating the Internet protocol from the Transmission Control Protocol. It was all smashed together. We had gateways that understood what it was that the hosts were sending. It knew how to break packets up and put them back together and knew how to route them based on Internet addresses, which is a name space above the level of any in a single net. That was the other problem that we had. Each network had its own name space, but it had no concept that there might be another net in the whole world. In fact, when you start gluing these things together, they have to have a way of talking about: "Which net am I?" And most of them didn't have any name for themselves, because they were the only one. So you don't have to name it if there's only one. Terrible oversight, but we never thought of it at the time in 1968, because there was only one packet net in the whole world, and it was Arpanet. So –

Pelkey: Now, this is early '73?

Cerf: This is 1973, early '73 when this is happening, and Bob was saying: "How am I going to hook all this stuff together?" And I'm saying: "Well, let me think about that for a while." So we developed this basic concept of gateways and an end-to-end protocol and a high-level addressing scheme, and it emerged in the form of something called TCP. We did three prototype implementations of it; one at Stanford University, one at Bolt, Beranek and Newman, and one at University College London. So one of the earliest participants in the development of TCP was Peter Kirstein and his group of people at University College London. And at Bolt, Beranek and Newman it was Bill Plummer and Ray Tomlinson who did the earliest work on the TOPS-20 system, only at the time it was 10-X, running on a KA-10. That was BBN's timesharing system. And we did it on an old PDP11-20.

Pelkey: And who was in your group at Stanford?

Cerf: A lot of people whose names you should recognize from the west coast: Judith Estrin, who is Executive VP of Bridge, James Mathis, who just went from SRI International to Apple to help build TCP/IP for Apple, Yogen Dalal, who built the Stars at PARC and then went to Metaphor Computer Systems as their VP of Engineering and is now VP of Software at Claris, Richard Karp, not the one at Berkeley but another one who was at Stanford and now runs a company that builds ISDN test equipment, and is sponsored by British Telecom, he's operating out of Palo Alto. Who else? Carl Sunshine who is now running the Western Research Center for System Development Corp. which is owned by Burroughs which is now part of Unisys. Who else? Darryl Rubin who ended up as an executive at Microsoft. Ron Crane who did all our hardware work

Pelkey: And Metcalfe was presumably poking his head in and out.

Cerf: Metcalfe was busy as a bee over at Xerox PARC, and the funny thing about our relationship is that in '73, we really thought we might get away with building a common protocol, but Metcalfe's people were gung-ho to make something work, and they had no one to convince that themselves about what to do. In the meantime, I had a big political problem on my hands, because every R&D group sponsored by ARPA wanted a finger in some of this, so it took me longer to get the TCP/IP protocols built and tested on the 30 or so different systems that had to be checked out than it did for Metcalfe to get XNS done on one kind of machine. So he went blasting ahead and –

Pelkey: On the Alto machine.

Cerf: On the Alto machine. So XNS happened, or at least the first version of it happened, before TCP finally settled down.

Pelkey: And do you know those dates?

Cerf: Well, I would guess that XNS -- the first set of protocols, reliable packet protocol, must have been done by -- I want to say '76 at Xerox, something like that, and I was only in my second version of TCP at that point. Third version came in '78. The final version of TCP-4 came about 1979, and standardized in 1980. Then went through an enormous number of implementations and tests and bake-offs and things like that among various operating system implementations until it was adopted by the Defense Department in 1982 as a standard. I've got documented Defense Data Network Protocol handbooks that SRI International publishes –

Pelkey: And what was the body that finally blessed it within the DoD?

Cerf: There was something called the Protocol Standards Steering Group, which was chaired by the Defense Communications Agency and had representation among most of the various military departments that formally adopted it. Before that, it was just something that had been standardized within what has now become known as the Internet Community. There was a body, never very formal -- the Arpanet standards always sort of got blessed by the old boys and then published as an RFC and, if Jon Postel had an RFC that said this is a standard protocol, that was it. Everybody went and implemented it. There's more recent history now that's even more fascinating about people and how standards are forming, but let's try not to lose the thread here. In terms of TCP development, there were four versions of it, and after Version 2 had been built and tested, we realized that we had to separate the Internet protocol part from the host-to-host part. The things that the hosts did to talk to gateways, and the things gateways did to each other, could be separated from what hosts had to do on an end-to-end basis. So we separated the functions of the TCP protocol into an internet function, packet format and addressing and routing protocols and error handling indications at the internet level were reported back using the IP protocol, and the TCP was used on an end-to-end basis for the hosts to talk to each other, and that's the part that did the flow control and the congestion control and the sequencing and retransmission and the like. That also made it possible for us to build other than TCP on top of the internet protocol, we built user datagram protocols on top of IP, and we built -- that sounds like Richard Carp.

Pelkey: Dick Karp has just joined us, and Dick you're where?

Dick Karp: I've got my own little company. It's called, as of this week, ISDN Technologies Corp.

Pelkey: ISDN Technologies Corp in Palo Alto, CA.

Cerf: And you're still funded by British Telecom, am I right?

Karp: No, they're less than half of our funding this year.

Cerf: Really? Ok. I hope that means that there was somebody else to pick up the other half.

Karp: Yeah, RBOCs and Bell Labs and people like that.

Cerf: Good, ok. Let's see, we just worked our way up to the point that the TCP protocols had split into TCP and IP.

Pelkey: So the third version was the one where the –

Cerf: Where the two were split.

Karp: I was long gone when this happened, so I can stay out of it.

Cerf: Probably the most significant difference between the third version of TCP and the fourth one, the one that was finally adopted, was that we played around with something we called "rubber end of line," and it takes to long to describe why that was even contemplated, but a lot of people raised a hue and cry about this concept and finally we agreed to get rid of it, so there was a great party celebrating the demise of the rubber EOL, and Version 4 basically stabilized. What's happened since then, since about 1980, was that the protocol was mandated for use in the military, and specifically it was mandated for use in the R&D community as well. ARPA really wanted everyone in its research community to be using that suite of protocols, so that multiple networks could be supported. It was essential to switch over to that collection of protocols to support all the local area nets that were beginning to come up in the campus environments, because they needed to be interconnected to a backbone, and the only protocols that would carry them through end-to-end were TCP. So in January of 1983, it was required that everyone operate with TCP/IP and related protocols, and we turned off the ability of the Arpanet to handle the old NCP protocol. There were mechanisms available in the IMP to cause it to reject, or not service, the old host-to-host, end-to-end protocols. We went through some real trying times to convince people that we were serious about doing that. Somewhere in the middle of 1982, we turned off the NCP capability for one entire day on the Arpanet, leaving the only people able to communicate were the ones who had implemented TCP. There was a lot of noise as a result, but it got attention. Then later on, somewhere in October, if I remember right, we turned it off for two days, and believe me that caused great consternation, because the electronic mail systems were still using the old protocols. People couldn't get their electronic mail, and that had become such an important part of people's lives, their ability to carry out research and coordinate things, that was very painful and there were a lot of people hollering and angry, but it was the only way that they could be convinced that this was a serious step, we are going to switch over to the new protocols, so in January of '83, we cut off all access to the old NCP protocols, with very few exceptions, people who pleaded special problems in getting their machine software up. So by '83, there was a very large community of use in the R&D world.

Karp: I still have a button from those days labeled: "I survived the TCP transition."

Cerf: That's right. I think I still have that thing. I think Dan Lynch is the one that put those together. Another name that you'll bump into in the TCP world, I think. What's happened since then, though, has been nothing short of remarkable.

Pelkey: One part was that you had to get an E-mail package running on TCP/IP that was at least the equivalent of what was running on NCP before you could get acceptance of it in the community.

Cerf: That's right. Absolutely true. We had to have telnet running. We had to have file transfer running. We had to have E-mail running, and we basically developed a new electronic mail protocol to replace the old ones that had used the file transfer system. The old NCP mail used the FTP -- actually it didn't even use FTP, it used the Telnet connection on the FTP. This is detail that you probably don't care about, but it was not very efficient. So Jon Postel and Dave Crocker, different Crocker than the one I told you about before (and is Steve’s brother), and several other people developed the new protocol called the Simple Mail Transport Protocol, or SMTP. They released a new RFC, like the ones I was telling you about before, and this one was 822, to describe the format for electronic mail, and there were other RFCs describing the details of the SMTP protocol, but that became part of the TCP/IP protocol suite. So you did need a complete package or protocols that were available and operating by January of '83.

Pelkey: And was that developed under your auspices at Stanford?

Cerf: No, they were developed outside of Stanford. Stanford only concentrated on the TCP and IP levels of protocols.

Pelkey: And you were still at Stanford?

Cerf: Well, I need to tell you that by about September of 1976, after four years at Stanford, I was asked to come to ARPA to manage the Internet project and all the other packet communications technologies, plus the network security program, so I did do that. I went to ARPA and stayed there until late in 1982. So it was really during the time I was at ARPA. My involvement now was on the management side, although I got to stick my fingers into the details wherever I had time, for many of the protocols that grew up around TCP. SMTP was one of them. But I don't take any -- I make no claim for intellectual territory on anything in those. I think other people deserve most of the credit and did most of the work. The thing that has caused the TCP/IP to take off as it has in the recent three-years is that local area nets have become so popular. On top of that, most of the equipment running on local area nets uses Unix as the operating system of choice, and ARPA, in its infinite wisdom, and I had nothing to do with this, chose to support Berkeley to standardize on a research Unix system that could be readily propagated at no cost to the government, other than supporting Berkeley, to all the researchers who needed operating system support with all the protocols built in.

Pelkey: When was that decision made?

Cerf: I would guess that the Berkeley -- I actually don't remember the date. Let's see, can we reason our way to it.

Carp: I know when TCP came out. It was first in 4.2.

Cerf: Berkeley 4.2. That has to have been in '80 –

Karp: Early '80s.

Cerf: Yes, it was certainly before '83, before we cut over, so it would have been '82-ish that that happened. Oh, I'm sorry, way before that, because it was before I left ARPA that the decision was made to support Berkeley to do the Unix work, and that surely couldn't have been –

Karp: That's going back to the '70s.

Cerf: '79 or '80, something around then.

Karp: There were versions of Berkeley Unix supported by ARPA before there were VAXs. For example, Unix was on PDP-11s, remember?

Cerf: Yes, so that would have been even as early as 1978. You know who would know the answer to that, the guy -- Bill Joy.

Karp: Or Fabry.

Cerf: Bob Fabry at Berkeley would know, or Bill Joy?

Karp: He did TCP, but he wasn't there when ARPA first started funding it.

Pelkey: He might know, though.

Cerf: He certainly might know.

Karp: He's the one who did the TCP on 4.2.

Cerf: So, the important thing is that the Berkeley release of Unix had TCP in it, and it was readily available, so it propagated very quickly all over the place, so the combination of a Berkeley Unix based TCP and local area nets being so popular just drove that protocol into the community. Not just the R&D community. It very quickly emerged as a commercial beast. So now there are over 150 vendors of TCP/IP based products. To find out the statistics about what's happening in the commercial world, the guy to talk to is Dan Lynch. Dan is president of a company called Advanced Computing Environments. They're out in Cupertino. It's a 408 number. 996-2042 I think is right. I can actually verify that. He published a newsletter called "Connections." In fact, here's the address. 21370 Vai Avenue, Cupertino, CA, 95105. So he can give you an idea of what the commercial side looks like. The estimates are our market for about $900 million worth of hardware and software by 1991.

Pelkey: Speak for a moment about OSI {Open Systems Interconnection}, how OSI, having used this layered model that had been developed through this effort, how that took a different tack.

Cerf: It didn't really take a terribly different one, except for the fact that it was strongly connection oriented. Let's see, the OSI work started largely with the efforts of a guy named, Hubert Zimmerman. Hubert worked with Louis Pouzin in France, at what was once called IRIA (Institut de recherché en Informatique et en Automatique). It has since become INRIA. We put a 'National' in there, because it's the national institute for research on information processing and telecommunication. It was Pouzin and Zimmerman and many -- a fellow that we had, Gerard LeLann, who was at Stanford for a year, came from that group. They all had been exposed to what the Arpanet was doing. They were part of the IFIP working group, but they were strongly influenced by, eventually by the PTTs. So the OSI work started out as an attempt to do a international standardization of packet communication protocols, but what Zimmerman noticed, and what I didn't want to see at the time, was that the standards bodies, which are largely PTT driven, and by this time who had become comfortable with X-25, didn't want to think in terms of datagram kinds of protocols. They were really very uncomfortable with that idea, and much more comfortable with protocol architecture that assumed that the underlying networks would do virtual circuits. So the protocols that they developed were very much more oriented towards the assumption that the underlying network would provide virtual circuits. So the X-25 version of interneting was X-75, which is the gluing together virtual circuits at the boundaries between packet nets. So the OSI work started out on the assumption that things would be virtual circuit oriented, and they didn't have an Internet layer of protocol at all like we did. It was really just imbedded down in the network system, hidden. X-25 is what the hosts saw as packet mode interfaces to the public network, and the X-75 was buried inside the networks at the interfaces between the public nets. Well, that's all fine. They developed it, the higher layers of protocol. They had a transport protocol, TP, and in fact, because of the community's disagreements about what should and shouldn't be done in the way of reliability and resequencing and the like, they had five different classes of TP: 0, 1, 2, 3, & 4, of which O, 2 and 4 have survived. Class o is just a copy of X-25. There's really no processing at all. Class 2 assumes X-25 service, but provides a little additional addressing and demultiplexing, but it doesn't have any end-to-end retransmissions or sequencing or anything like that. Class 4 is very much like TCP, different in some details, but it had all the retransmission and sequencing and window based protocols that TCP has. The OSI people then developed some additional –

Pelkey: Excuse me. What time frame is this now?

Cerf: Ok, the original OSI work, it seems to me, started about 1976 or '77. I can remember walking down the street in Geneva with Zimmerman somewhere around '77, talking about what was happening in the -- what his plans were for the standardization of these ideas, and he was telling me he was going to start out with virtual circuit oriented stuff, because it was the only thing that he could sell in the architecture. He knew, personally, I don't know if he publicly admitted it, but I think he knew and said so privately, that he wanted to introduce the datagram notions, but it would be later, after everybody was comfortable with the architectural model based on virtual circuits. He was much more politically astute than I was at that point. So the OSI work developed largely along virtual circuit oriented lines, and they invented a lot of protocol functionality and layers that at the time I had trouble understanding the motivation for. Things like a session layer I kind of understood might be useful, although we buried it in the TCP functions.

Pelkey: Presentation Layer?

Cerf: Presentation layer was complicated, and I didn't understand what it was doing there. I'm more sanguine about it now, but only because it's driven by some applications. I always felt that all the R&D work that ARPA supported, and subsequently other American research agencies supported, was all driven by a need for a particular outcome. "We have to get X done. What will we do to do that," as opposed to: "here's a wonderful architecture and we stuck in some concepts. Now let's see if we can build protocols around it." So I was not very warm to the OSI protocols and the architecture above layer four, because I didn't understand exactly what was driving those choices. By '79 or '80, the OSI ideas were starting to get considerable publicity, and people were finding the ideas attractive. I always thought that the biggest contribution that the OSI made was to provide a common vocabulary for having disagreements. In a sense, that's important. People can't discuss what it is that they are concerned about unless they have a common language. So you find yourself in technical arguments struggling to make sure that you're using common terminology first, before you can have a genuine discussion about concepts, and to resolve any difference, or at least to go away understanding that you don't agree and exactly what it is that you don't agree on. So the OSI stuff is still, as you know, remains to be fully delivered. There are a lot of people putting a lot of time and money into it. There aren't very many available commercial implementations of it, and the testing of this is going to be very time consuming. It needs so many parallelized tests to make sure that every vendor's software packages –

Pelkey: Works with everybody else's. Jumping ahead, there is talk of the government replacing TCP/IP with GOSIP which is the –

Cerf: GOSIP is the Government OSI Protocol specifications. This is the document that, in theory, you can include in an RFP. We need to wait for a (unintelligible)? No. So the government has a belief that, if you could just buy this stuff off the shelf, it would be less expensive than having to maintain special packages that were built for the government. I don't disagree with that at all. I think that's quite true. The only thing that's mildly disappointing is that the government jumped onto this GOSIP idea after the TCP/IP is now fully and commercially available from most of the vendors. So they may even be shooting themselves in the foot if their time frame is such that they insist on the adoption of the OSI protocols before they have been fully tested everywhere for inter-operability and for serviceability. I think over time the vendors will, as they already have, conclude that they really do want to have international standards so, as a vendor, they don't have to build 16 different protocols. So there really is reason to move to the OSI protocols. I don't disagree with that at all. I have some concerns about when and under what conditions to make such a move. I think that the vendors recognize the utility of an international standard. They also recognize that they have an installed base. I don't think the government has completely understood that it has an installed base, but it has a very big investment already in the TCP protocols, and their users are heavily invested in it, and their transition from wherever they are to something in the way of OSI is going to take some careful planning.

Pelkey: Let me go back to UCLA. There's a Chinese professor who's name I forget –

Cerf: Wesley Chu.

Pelkey: Thank you. Who is credited with having a patent on the statistical multiplexer?

Cerf: That's correct.

Pelkey: Was he associated with any of the things that were happening in any of the things that were happening with ARPA?

Cerf: Rather peripherally. He was rather peripherally involved.

Pelkey: So his ideas -- the time frame was roughly the same -- were independent of –

Cerf: Completely independent. He came up with that notion -- certainly wasn't party to any of the internal designs of the network. Certainly was aware of it as everyone else was, but he certainly deserves full credit for having come up with that.

Pelkey: Because you used multiplexing in the IMPs.

Cerf: You can certainly view that as a kind of statistical multiplexing, except that the IMPs were doing an awful lot more in terms of routing and flow and congestion control. They were relying on more than just having a buffer source.

Pelkey: It was a networking as opposed to a link protocol.

Cerf: That's right. So, Wesley's idea -- clearly has been adopted by a lot of equipment vendors, is applicable to certain situations, but not to the general networking environment.

Pelkey: It's interesting that these notions that developed out of networking – datagrams, packet networks, really haven't found their way into the marketplace as strongly as has that technology through the local area networking, which originally was started for the telecommunications, i.e. the wire and wide area network application, and it has found its home more or less in the local area net, intra-building communications.

Cerf: I think probably the way to understand that is that the commercial side has moved down the X-25 virtual circuit path, but was unable to cope with the kinds of networks, like local area nets in which there wasn't any virtual circuit mechanism available, and at the point where the guy with the local area net wanted to talk to somebody else with a local area net, or just wanted to have hosts talk to each other, the X-25 was not a suitable protocol for them. Otherwise, they'd have to go and build all of X-25 or some end-to-end virtual circuit thing in the host, because the Ethernet didn't have anything in it. It was just a cable and a transceiver. Well, they didn't bother to do that when they discovered they could buy TCP or get it for free from various and sundry places. Since most of the local area net hosts started out on Unix-based machines, it was natural for people to use the TCP, because that came with Berkeley Unix anyway.

END OF INTERVIEW

