Computer History Museum

Interview of Robbie Forkish

Interviewed by:
James Pelkey

Recorded July 7, 1994
Redwodd City, CA

CHM Reference number: X5671.2010

James Pelkey: We had been talking. You had joined Tymnet in 1978 from having first been in another front-end datacom company.

Robbie Forkish: There was -- the company I was in had an interesting approach, because it was in the days prior to communication architectures, and there was a lot of telecom software running around where the applications and the management of the terminal characteristics were interspersed in sometimes alternate lines of code. You would do your banking logic and then you'd say: "Well, if it's an IBM 1060, yeah, send this kind of character. If it's an NCR 270, send this, and so on and so forth, right down the line." And, people being overwhelmed by the ability to add new capabilities. From a competitive point of view, banks were out there trying to get new business, new customers, and employ new technologies. So what we did was very simple-minded. We created a front end processor for financial data centers that, from the host side, emulated a mag tape, provided a standard format of messages to and from the host so that the applications programmer, at that time, I think, perhaps uniquely, was free from concerns about individual terminal characteristics. The technology was interesting; the business did a lot less well than perhaps it might have, but it was only when some of the banking applications expanded to having network among themselves, and having banks that played the role of switches, that I became interested in networks and, in this case, the networking of multiple networks together. As I learned more about the industry, clearly there was the emergence of the value added networks at the time, and Tymnet had some appeal to me as I went out there in '78. Tymnet had a very interesting technology. It was interesting in several regards. One, it was very much home grown, and two, it was designed without ever having the intention of growing into kind of a generic communications network. The idea was more to support the core business of Tymshare. Therefore, the network was terminal oriented. It understood that sessions would have a predominance of data coming in one direction and a light amount of traffic going in the other direction, for example, as is characteristic of timesharing sessions. The technology evolved, of course, as there were business imperatives, particularly in the area of first building the VAN itself and second, the marketing of private networks which started to occur in the early '80s. It was, I thought, quite interesting, particularly after spending so much time convincing people to get out of their private network orientation to go to a VAN, to now have a private network solution, but it was a technological sell. We were selling against modems and statmuxes, and we were selling against people having to manage a large number of things as opposed to a smaller number of things, albeit more sophisticated, more expensive, and so on and so forth.

Pelkey: So, in the early days, you saw that you were, in fact, competing with modems and multiplexers. That was where your competition was.

Forkish: Very much so. Very much so, just as different approaches, you could have a bank of modems with a router to dial into a timeshare host from all over the country and get an 800 WATS arrangement, or you could have the virtual capabilities of the value added network, or you could buy a stand-alone private network, and Tymnet was in the latter two of those three options.

Pelkey: What was happening with the IBM world at this point in time? Did it impact on you?

Forkish: The IBM world was not where we focused, because of primarily the synchronous

nature of the IBM protocols and the asynchronous characteristics of the business and the hosts that we supported. Over time, though, Tymnet developed capabilities that they called Tymcoms, but they learned better and called them Pads, because we had an official name for them, and they were simply able to take originally bisynch frames and convert those into logical packets that float across the Tymnet network. There was such a big market that there was clearly an impetus to go do that.

Pelkey: And you don't recall what time frame that was that you did that?

Forkish: That was probably going on in the late 70's, although I think that there was a reluctance on the part of com managers to go to what looked like a kludgy approach, but with X.25 and the emergence of pads, the emergence of bisync pads, and so on and so forth, the concept started to become more standard. Then, I think people are more willing to take that kind of risk. It's a -- from a development point of view, you're never caught up, because you're always chasing the latest changes in protocols and so on and so forth, so it wasn't intrinsically satisfying, but appeared to be pretty good business areas.

Pelkey: And Telenet was --

Forkish: Telenet was the very clear competition.

Pelkey: And Telenet and Tymnet -- you competed with each other frequently?

Forkish: I would say -- again, I was in development, so I didn't have firsthand evidence, but it seemed to me that well over 90% of the time, it was head on head competition with Telenet. This is before the days of Uninet and Graphnet and some of the others that came on, and probably even before Northern was selling their data pack equipment in the US for private network solutions which they actively do today. So, I think that the world looked very bipolar to us, and it was also bipolar because Telenet had all the technological credibility. They had the people from ARPA; they had written all the papers that were published; and they had the right way of doing it, and Tymnet, of course, had some other way which, by implication, was not quite so right. The truth of the matter is, I think they both had their place. Tymnet was less a packet network; it was more of a network of statmuxes, however it was not advantageous for Tymnet to market itself that way, so it was always the packet switching alternative. It was connection oriented versus connectionless; it was centrally directed versus having distributed control; it used different types of processors. They were just as different as can be, and the competed head on head, so it was quite a thing to see the debate and, at the time, with the emergence of LAN technology, there was a similar debate on broadband versus baseband, and it looked like, gee, every new market's going to have some bipolar arrangement, and sure enough, VCRs came out later with VHS and Beta. So I think that all of these issues came to some resolution that perhaps there are some applications where one technological approach is superior and some where another technological approach is superior, and you can't say that one technology is simply better, that there are a number of trade-offs. So, if I walked away with a lesson from Tymnet, that was probably it. However, I learned a great deal about networking as part of the team that implemented Tymnet II, which was Tymnet's movement towards distribution of network logic of functionality. Instead of being purely centrally directed -- maybe it would be more correct to say instead of being

purely centrally controlled, it evolved to be centrally directed, so the central process, called the supervisor could, instead of assigning all the resources for it, would tell it to build a virtual circuit, and therefore off-load much of the processing that was required. That was important for Tymnet because the technology was originally designed to support perhaps 100 nodes in the network, and as we were growing up in the 2,000 range, it was showing signs of falling apart. So, our two choices were: one, to take this supervisor that ran on an old XDS 940 Honeywell, to maybe convert it to a Cray or something and just continually try and add horsepower to it in a forced fashion; the other was to find a new network architecture. It was the latter approach that was taken. The next step for me was going to BNR and the intrigue there was --

Pelkey: And you went to BNR in '82?

Forkish: I went to BNR in '82 in the Mountainview lab. They were working on their next generation technology in their office automation area. I was interested in voice, number one, because even then, there were people who would look at our private network solution and say: "But I'm spending 80% of my dollars on voice. What are you going to do for me there?" At the same time, I felt that, perhaps not so correctly so as it turns out now, that the PBX was going to play a role competitive to the LANs, and BNR seemed to be doing some very interesting work in that area. We were working on packetizing voice but using twisted pair distribution to the individual stations. I probably could have stayed there and designed and developed for quite some time if the NET opportunity hadn't come along, which was effectively, I think, towards the end of '82, I was talking to Roger Chrisman, Audrey and Sarah Schlinger about NET, although it didn't have a name at the time. My name for it was Crazy Startup. The date I decided to jump ship, so to speak, was Thanksgiving of '82. I remember very clearly, Roger and Sarah came over for dinner and my wife and I had been wrestling this one for a couple weeks, and we finally decided, sort of as a result of that, that I was going to make the jump. Things --

Pelkey: That was a big step, to go from a salary and something you like doing to this notion.

Forkish: Yeah, I don't claim any great intelligence in having done that, although it was one of the best decisions I've ever made, not coming to NET necessarily, but by January it looked that the financing would be imminent and that this new company was going to get off the ground. We were getting a lot of interest on the part of venture capitalists, and we had some critical meetings, so I -- some critical meetings with investors, so I decided to go ahead and anticipate a successful meeting and resign my position with BNR.

Pelkey: Which you did when?

Forkish: I think February first I was unemployed. We didn't get funding until August, so, in my attempt to beat the investors, I did a good job of that, and I was part of the team. The reason I say that was such a good decision is one of the things that happens in software systems is you bring a number of designers together and it can turn into a philosophical debating society and it never gets off the ground. Although we were frustrated and we thought we should be funded, we were ready to go to work and, in fact, in some danger of having to pack it up because of no salary and no resources, I had nothing better to do than work on the design of the product. There was a tremendous technological challenge to me in building a network circuit switch, and I thought I

knew how to do it. I had learned enough about networking principles at prior locations. I used to say that I learned networking at Tymnet, and I learned how to manage software, large software programs at BNR, and I was glad to have that combination of disciplines. By the time, in August, that we actually got around to hiring software designers, making offer letters -- they started the day after Labor Day in '83 -- I had close to 50 pages of architectural documentation and functional specifications written, not that I am a one man rocket scientist that can design the best networking technology around, but more to the point, there was consistency throughout the design. It was a model that people could buy into and, by the second week, we were doing detailed module designs, so there was a team of software people that had all worked together before. We really hit the ground running, and we were also, I think, had the luxury of doing things right in the sense that we wanted to do them, we didn't want to take shortcuts. We wanted to resist the urge to simply start coding. Many times my brand new managers would come around and say: "Well, have you started coding yet?" with some look of great concern on their face, and I would say: "No, no, we're getting there." In fact, except for some experiments that we coded up on our development system, we had done no coding as of January first. Now there was some real concern. We actually coded the entire system in about six months. It was one of the most satisfying efforts I've been involved in. It was highly focused and well directed, but we had a charter and a goal, and we had developed precisely, or at least reasonably close, to where we thought we'd end up. Certainly we had some mid-course corrections, we had design decisions that were based on poor assumptions as we found out, but we were not designing on the fly, and I think if our software is of any value today, it's largely because we resisted taking the shortcuts. We had a long term plan, we worried about maintainability of the software as well as functionality of the first release in the early stages, and, at the same time, now going back again to early in '83, Roger Chrisman, who was the prime force behind the California team getting together until we met Bruce Smith, Roger was designing the hardware, and he and I together were designing the architecture as a hardware and software work together. So, much of what I said about the software is also true on the hardware side. More importantly, the hardware and software teams as they were, one person each, were spending so much time together that there was an intimate relationship between the design of the hardware and the design of the software. I met -- when we were trying to build the team, one of the people we interviewed was Al Zucchino, who is currently VP of Operations at Tymnet. We needed a VP of Marketing, he wanted to be president, and we didn't have a meeting of the minds, and ultimately he went off and started NSS. I saw him at a trade show just a few months after we had started shipping product. Now, they were six months behind us, in terms of the founding of their company, and I asked: "How's it going?" And he said: "Well, just terrible. Hardware hates software, software hates hardware, and they both hate marketing." So, at least we had, in large measure, the hardware and the software people working together, although we might have shared, in kind of a tongue in cheek sense, the same sense of marketing. Part of the reason for that was, although I've always felt that NET was a marketing driven organization, most of the product definition came from the engineering side and came from engineers who had good market sense, who had talked to customers, who had been in the industry for a while, who knew about maintainability of product, and so on and so forth. So, we didn't shortcut the marketing process, but we didn't have non-engineers involved in product definition at arms length. In fact, our first VP of Marketing didn't start until we were some number of months into the program, and, although we had occasionally some loud meetings between engineering and marketing, I think, to the benefit of the company and of the way the product looks today, we had some very opinionated engineers

who had some strongly held beliefs and who were very forceful in their thinking. We agreed to disagree in many cases, on many occasions, and I think we were able to make a better product as a result of that.

Pelkey: Did the software engineers -- when you hired software engineers, how many did you hire in the first three months or so?

Forkish: We had, by Christmas time I guess, a team of about seven people.

Pelkey: How many of them had you known before or worked with?

Forkish: Five, from Tymnet and from BNR. In some case, from both, so we had some sense of continuity. Although we didn't work together as a team in any of the prior locations and we had some very real personality clashes, we understood how to work together. We understood how to -- you know, what each other's strengths and weaknesses were to work around those to some extent. It was -- in the first year, I was not at all a manager, I was very much a leader, if you can draw a distinction between those two. I was as much leading the 'charge of the light brigade' as anything, and that simply set up circumstances that can never be duplicated except in the environment of a start-up and creating something out of nothing.

Pelkey: We talked about connection and connectionless. Were those important issues at this point in time?

Forkish: They were pretty important, but we never got stuck in the mode of debating those because we had largely fleshed them out before we had begun the recruiting process. The 'connection versus connectionless' was really less of an issue because, in a circuit switch, it seemed pretty obvious that we would build connection-oriented sessions. We spent a lot of time early on, having come out of BNR with its resources and maybe its penchant for taking technological risks which, I mean, in a start-up, it's ironic, but we were very conservative from a technological point of view. We wanted to be innovative, but we wanted to use proven technology in being innovative. So, we looked at what today would be called 'fast packet' as an implementation. In fact, the very first internodal protocol was called 'time division packet switching,' which we named, and the idea was to be flexible so that we could packetize speech and other forms of information going across T1s in a way that was flexible. It was one of those wonderful designs that, when you sat down to try and implement it, you found it was really not implementable, and our concern was very much centered around cost and reliability, and we thought we would be burdening the system with too much cost and taking too many liberties with what we thought were reliable and proven practices and technologies, so ultimately we went with largely a standard approach and, in fact, the time space time architecture of our product had been around for some time in central office equipment. The other interesting issue, though, was in centrally directed versus distributed network control, and this was an area where I felt very strongly and where I was probably the primary architect, given my background and that it was largely a software task in its development. All of my experience being centrally directed networks, some people were both concerned and surprised that I took the decision that we should be distributed in our approach, but we all agreed, as we knew from our days at Tymnet, it was easier to sell a distributed approach. It was easier for people to buy into that. There wasn't some

ugly thing that was going to burden the price of the network that you had to start off with, and there was no single point of failure from a network point of view, nor was there a bottleneck for network processing. We felt pretty strongly, although we debated very early on 'should we come out with a very simple product with static routing or should we go with our best shot and come out with a product that has alternate routing?' We knew we wanted to come out with a minimum feature set and have a phased approach towards getting functionality to the marketplace, but at the same time there was going to be a critical mass that we needed in the initial product to differentiate ourselves. Timeplex, as of the beginning of '83, had just announced their Link 1 product. There were people who thought that we should simply pack up our bags; Timeplex was going to walk away with this market and we shouldn't even try, and yet we saw a lot of weaknesses from a technological point of view in Timeplex's offering. We went and attended a briefing in Santa Clara after they had made their announcement, as they were describing to some of their customers, we happened to get invited, or got invitations rather. They were describing what their product was going to do and be like. As we sat in the room, we noticed that people asked them questions about T1. One of the questions was 'How are they going to treat red and yellow alone?' Very standard, very standard in T1s. Not only did they not know the answer, they didn't understand the questions, and it was real clear to us that there was an opportunity whether Timeplex had a Link 1 or not, and it was that meeting, that Timeplex briefing, that we sort of became very encouraged again. We had a new resolve. We didn't think of them as a threat, so we charged right off.

Pelkey: How about that.

Forkish: The thing that I felt very strongly about was not only that we should come out in the first release with rerouting capabilities, but the rerouting should be fast enough to largely make a failure in the network transparent, and, looking back, although there was some tremendous incremental complexity in the product as a result of doing that, that that was probably the differentiating feature that lead us to the market share that we have today early on. Now today, that's not necessarily a buying decision, but early on we talked about that at great length. We gave demonstrations, the most impressive of which had two brokers talking on a phone and we were able to break a T1, have them be rerouted, and they didn't know it had happened, and they were both from New York, which meant they were both talking at the same time, and we were able to demonstrate time and time again that the difference between fast rerouting and slow rerouting is not necessarily one simply that's an incremental measure, but it's the difference between a conversation being disconnected or continuing. It's the difference between a host to host session having to be disconnected, go through restart procedures, or having a momentary pause in its transmission, and I think it's that concept and what we ultimately dubbed 'Applications Availability' that became the key differentiator for NET.

Pelkey: What was it like when you got your first product installed?

Forkish: Well, you always have these interesting milestones, and most of senior management at NET had come from hardware manufacturing, hardware design background, so the software people were a bit of an anomaly, both in terms of the discipline and in terms of the culture. I remember one of our early investors visiting and came back to me about the software people, she wanted to see who was back there, and she told us a story about this other group of software

people who had found a bird with a broken wing and come in and nursed it, and it was 'Surely we would be interested in this. We were software people too.' We had -- as software people do -- we drew all kinds of diagrams, had boxes and arrows, and we could never find a white board big enough. Well, we had a huge picture window in one of the conference rooms. We took our markers and we wrote on it, and people expressed concern about security. You know, we said: "Gee, they'd have to read backwards and this and that," so we kept up that practice, and even today there are references about the fact that the early software team used to run around writing on windows. It's another form of abnormal behavior, so we were, perhaps, a little bit different. The greatest compliment, I think, I've gotten in the five years at NET was from one of the other hardware designers, where they had a mid-course correction which caused them to change the hardware design, and about six months later, as we were just getting out beta product together, we said: "You know, I worked on a -- in my prior job -- on a system where the hardware was a month late and it made the software be two years late, and here where the hardware was a month late, it only made you be a month late, and I think that's really great." The other thing, I think, was we were, because of mature seasoned management and, because I'm talking technology I'm not giving credit in the company for Bruce Smith and the management team, it was -- what he did that was so significant was we always worked two schedules, and he had an outside schedule which showed that we would be complete and be shipping a product in May, and there was an inside schedule which was named 'the insane schedule' which showed that we would do it far sooner than that. It showed that we would do it in December. We were off our insane schedule by a month; we delivered in January, but to the outside world, we came in five months ahead of schedule, and to have gone through as hard as we did and worked as intensely as we did for 15 months, to come out at the end and to have missed it would have been a very unsatisfying feeling, but to have world that we acknowledge that we came in early and that's never happened before and so on and so forth really added to the satisfaction of the experience.

Pelkey: I would imagine.

Forkish: But, there's nothing like seeing your product in action for the first time.

Pelkey: And what was that like?

Forkish: We took a number of engineers to New York City to install Bankers Trust, which was our first major beta site, and some number had not ever been to New York City before, much less participating in the type of experiment or business event that we were participating in, and we all had a great time, we had a great time with the customer. There were a large number of things that didn't work, although there were a large number of things that did work. There was, as is the case in any of these milestones that we had, there was mass hysteria over the things that didn't work, and 'Oh, it's never going to work, and we might as well fold up now,' and so on and so forth, and, you know, from the engineering side, you look at that philosophically and say "Well, it's going to take us about two weeks to fix it. Please calm down." So we were able to get through those, but I'll never forget that experience, just never forget that experience.

Pelkey: I would imagine.

Forkish: In fact, one of the engineers went out and said: "I'm in New York. I'm going to ride on

the subway." We were going from one place in Manhattan to another, and he said he knew he was in trouble when he went over a river. He was seen about two hours late getting to the meeting, but it gave us a lot of fun. It gave us a lot of fun to continually remind him of that.

Pelkey: So the first beta went well.

Forkish: We were very fortunate early on in having, as I guess would have to be the case, the customers by their nature were risk takers and they were also very supportive. They had some of their own prestige on the line of course. We had worked very hard and had a lot of pride in our voice processing capabilities and our voice compression was also a key differentiator. In fact, it was internally integrated to the system, but there were no standards for ADPCM at the time, but there almost were, so we decided instead of waiting, we were going to go with a proprietary approach. Well, there were concerns about what quality would be like, so the folks at Bankers Trust did what they called their Pepsi test. They sent a note through all their user organizations that said: "Starting in two weeks, on a Tuesday, we're going to convert from PCM to ADPCM. Please let us know what you think," and then they did it the next day. Sure enough, two weeks later they got complaints about 'the quality is worse today than it was yesterday,' and they said: "Oh, by the way, we did it two weeks ago. We'll see what happened last night." So they solved the problem of managing their own users in a very clever and creative way, and it's interesting that the end user and the vendor had to be creative working together to make the installation a success.

Pelkey: That's a great story. Now, during this period of time, there was Cohesive and NSS and there was Timeplex that was out there. Principally, there were the other two start-ups in which the group of you knew the people, because they both came out of Tymnet -- some of the people came out of Tymnet at those organizations as well.

Forkish: That's right, in fact, all three, you could argue, had their genesis out of Tymnet; Art Caisse in founding Cohesive and Al Zucchino in NSS. Four of the six founders of NET were from Tymnet.

Pelkey: Right.

Forkish: We had, I think, respect, fear, and dislike all at the same time for Cohesive, because we knew we were going to compete with them. From day one, when we got our funding, we knew that our competitor was going to be cohesive. No one ever really took NSS seriously -- some good people there. Some of the better ones, I think, left early and had their own new venture which went very well, but we always thought that the enemy was Cohesive, and I think both companies probably made it into more of an enemy type perception than they had to, but it was also part of our feel for motivation and so on and so forth. I had always remembered the story about Bobby Kennedy, that he would have his chauffeur drive by the Teamsters building at night on the way home. If the lights were still on up on the 11th floor where they were, he'd tell the chauffeur to take him back to his office. He was going to work later than they did every night, and I think if the Cohesive offices had been on the way home for us, we probably would have had very much the same attitude. We simply were not going to be beaten by those guys. We didn't think that Timeplex represented a threat at all, and we weren't sure that NSS was ever

going to get a product to market. So, one thing that you can get from this is that probably by necessity, we had to be a little bit mentally arrogant, otherwise we would have simply gone insane. We had to assume that we were the best act in town. We had to assume that the other guys were going stumble with this, that, and the other thing, that their approaches were flawed, and so on and so forth, and I imagine the reciprocal was true on the other side.

Pelkey: Did you have insight into what Cohesive's structure might be, how their product might be different than yours?

Forkish: No, very little, although we did feel that there was probably more concern on their part about understanding what we were doing and directly competing with us in the marketplace. Some of our early successes, in fact one said: "You know, you two guys are even, but I'm going to go with NET because when the Cohesive guy gets in here, he spends his whole time talking about NET. When NET gets in here, you spend your whole time talking about NET, so NET must be the right thing to buy." He said he could never get out of Cohesive what was better about theirs, only what was worse about NET's, and I think their negative selling hurt them. I was not going to be the one to tell them that because we were reaping the benefits from that, but it was -- if there's a need to be motivated, there's also the possibility that it can go over the line, and maybe they were so intensely focused on beating NET that they did themselves some damage. They have a very fine product and some of their tactics, and I think maybe the side effects of their culture visa-a-vis NET, might have been negative to them, but they --

End of Tape Side

Pelkey: Why do you think it is that Tymnet was the source of these three companies?

Forkish: There are, in my mind, two reasons. One, quite frankly, I think that Tymnet was not as strong as a group of managers as might have been true in some other companies, and therefore forced many people in the so called lower levels of the organization to have to take on tremendous responsibility, and it created an environment where most people failed, but the ones who were successful were successful in the presence of great challenges, and could effectively test their mettle by having more and more responsibilities assigned, and as cream rises to the top, growing in the organization. That may or may not be a major aspect of it. The second thing was Tymnet's involvement in the Caravan experiment in early '82, and one of the things that the Caravan showed -- Caravan had to do with the wonderful opportunity that DTS was going to present to a number of companies and everyone was going to be making money hand over fist.

Pelkey: What's that, Digital Termination Service?

Forkish: Yes. The idea was that satellite technology could handle the wide area portion of transmission networking, and that the last mile had to be solved and DTS would solve that, but in between the long haul and the last mile, there needed to be some switch, and the Caravan experiment made it very obvious to a number of people at Tymnet that there should be this product and there isn't. In fact, here was this very sophisticated capability that had as its switching mechanism phone calling someone on the top of a building and telling him to move the jack to a different plug, because that's how we set up 9.6 circuits. There had to be a better

way, and the thought that there had to be a better way and the thoughts that people had who were technically oriented lead to the idea of T1 multiplexers, T1 being the most primary and attractive transmission vehicle for satisfying that demand on the wide end. The idea is that maybe through DTS on the back end or some other form of distribution, there would be the opportunity. It turns out our market never was DTS, it was selling into the large end-user organizations, and so it's a funny thing because we might have gotten started for the wrong reasons. We were all chasing a DTS opportunity that needed a particular type of a switch playing a role in the middle, but when all was said and done, DTS itself withered on the vine.

Pelkey: If I remember correctly, when you first came out, you were more T1 to T1 switching, and it took you a while to build the cards up to be able to bring low speed in. Is my memory correct?

Forkish: We had low speed data capabilities, but what was unique was that we brought in voice on a T1. We didn't fool around with the individual channel analog interfaces as all of our competitors did. We knew that if we did that, once we designed one, we'd have to design the other 83 that went with it. It was much simpler for us to design DS1 interface to something like a channel bank and then let ITT and other channel bank manufacturers worry about the 83 different types of channels to terminate. They all look standard to us, so we were truly a digital switch. We didn't have any analog components. Now, the other thing was, our own expertise allowed us that decision. We didn't have any analog designers, so it was kind of a very pragmatic thing that was one of the factors that lead us to a strategic decision which, in hindsight, turned out to be a good one because it lead to us -- a perception of us being a T1 switch as opposed to a T1 multiplexer.

Pelkey: Yes, and as you were saying, the genesis of that idea was really seeing this satellite, which was presumably going to be high speed circuits, and being terminated at the top of a building or something, and then you'd have this DTS link up, but there was this gap in between and you had to do this switching between these end points and whatever the channels were happening off the satellite, and knowing how to get the traffic from one to the other. There just wasn't any kind of a natural switch that sat at that point in time.

Forkish: That's right.

Pelkey: And that Caravan --

Forkish: Caravan experiment, now that was the name that Tymnet used, I think it was the name that other providers used too. We worked with SBS and I think LDD was one of the distributors, a M/A-Com company. That's actually how Roger Chrisman got involved in this. He came out to Tymnet to work on the Caravan experiment working for Art Caisse and --

Pelkey: Where was Roger before that?

Forkish: He was at M/A-Com in Maryland.

Pelkey: I see. So he joined Tymnet?

Forkish: He joined Tymnet and the Tymnet team made a proposal to Tymnet senior management that we should go build a switch like that, and they thought that that was not in their best business interest, and there were a number of people who evidently disagreed because they formed startups to charge off after that opportunity. It's almost ironic now that Tymnet has an OEM agreement with DCA Cohesive to do something that was proposed for them to build some number of years ago.

Pelkey: So there was this experiment that went on that really, those of you that knew there was a better solution, having been prepared through your prior experiences and then to see the problem, and at the same time, I guess, there was this macro stuff that was happening in terms of, around this time --

Forkish: Now that's interesting, because that's the perspective that Bruce Smith came from. He saw the shifts in the industry and the trends, and he came from a very macro point of view, and my own point of view was, 'Geez, there should be a product that looks like this and there isn't, and I know how to build one, and it seems like someone ought to fund us so we can build that. I'm sure someone will buy it.'

Pelkey: Did you have any in to that market or any of that at that point in time?

Forkish: We were doing a lot of work and we were looking at market surveys, we were sizing the market, we were trying to figure out a market share standard approach, but we were also interviewing customers and saying: "If we build something like this, will it serve your needs?" And they said: "Well, if it had this kind of capability then it would," and so we did both the top down and the bottom up, I think, assessment of what the requirements were. Our sort of final act after Bruce came on the scene was the investors wanted to hear from customers that they might buy something like this, so we had a very interesting team that we sent to the east coast. Audrey was, at the time, eight months pregnant. Bruce Smith, our president, had just hurt his back. So there's Walt Gill, our VP of Engineering, carrying bags for three people; a pregnant lady, and a hobbled over CEO coming in to talk to people about this wonderful new technology they're going to turn into a product. It must have made quite an impression.

Pelkey: When did that happen? Did that happen after the financing?

Forkish: That was -- we had gotten some seed money, but that was a precondition for the full first round financing.

Pelkey: So that was like in July --

Forkish: That was in the July time frame, it may have even been in July.

Pelkey: That's a great story. The new world cometh.

Forkish: Yeah.

Pelkey: Walt certainly must have thought it was new worldly.

Forkish: Walt's got a great sense of humor and he's the only one who could have pulled that off, I think.

Pelkey: That's a great story. So Tymnet happened to be doing this Caravan experiment that was the catalyst, if you will, that you sort of realized, 'Wait a minute, this is crazy. If this is all going to happen, a product is needed there. Some functionality is needed.' But it was Bruce that brought this, from your perspective, this issue of 'Wait a minute, here's what's happening in terms of market dynamics.'

Forkish: Much more so. You know, we were thinking about that, but when an investor would ask us: "Are you going to work with AT&T or compete with them?" We'd go: "Well, gee, I guess we could do either," where Bruce would say very confidently: "We'll do both," and it was just -- it was the difference of perspective, I think, that allowed us to sort of create our own identity. Sort of an interesting side note in terms of identity was our investors must have been very frustrated because we couldn't come up with a name for ourselves. Originally, Roger and Sarah and Audrey had the name PCM, and that doesn't stand for Plug Compatible Manufacturing or Pulse Code Modulation, it was Packard, Chrisman, MacLean. When we got our first round of financing -- the seed money, rather -- we missed our deadline for names. The lawyers made up one. We were called Metrolink for a while, and it was like we got an extension on the deadline, 'Ok guys. You're going to build a company. Can you please come up with a name for it? Aren't there any other pieces of fruit left or Greek gods? You know, come on.' And we had -- you know, we had these dinners, we tried all these ways of being creative, but there was nothing we were comfortable with, and finally it got down to the 'well, what are we going to build,' and so on and so forth, and the name seemed pretty ho hum after all the ones we had looked at and all the searches that we had people go through, and so on and so forth. I thought our finding a name was going to be the lawyers' full employment act for one thing.

Pelkey: So then, you got the product into the marketplace and things were beginning to go well. In terms of the dynamics of the marketplace from the technology perspective over this period of time of what must be, what, '84?

Forkish: Yes. We were shipping product in earnest -- January '85 is sort of the date we acknowledge now as when we started shipping production product.

Pelkey: What were some of the technical issues in terms of architecture, not specifically within your box but issues that were happening within the data communications industry, issues of LANs being connectionless and where you sat in terms of increasingly being put in the position where you're having LANs and LANs coming together and the role of network management and this backbone utility network? What kinds of demands were those making in terms of changes that you had to respond to on a technical level?

Forkish: First, those were almost largely '87 and '88 issues. In '85 it was a little bit different. Here customers -- died-in-the-wool telecom people, switch makers, and carriers -- were probably just turning white over the fact that there were all these ignorant datacom people coming in and

trying to manage transmission facilities, and the number of vendors who didn't understand things about how alarms are encoded and pulse density on the T1 and delay and its impact on echo and so on and so forth, it was just craziness, and largely what we were selling was our own credibility, and when one of our early customers would call AT&T and say: "Your T1 is broken and I know because my two IDNXs can't communicate," AT&T's reaction was largely, I think, to laugh and say: "We're not surprised. Those two things can't communicate, and we never thought it would work anyway." So typically, we had to put an awful lot of our technological investment in supporting the support process, and it's kind of mundane in terms of technological trends, but in terms of the trends of network management, the service orientation, the expertise bottleneck and so on and so forth, it's probably a pretty important dynamic. The second issue was capacity. The Link/1 was considered a Big Kahuna because it was reported to support six T1s, and who would ever need that many? Just think of the amount of data you can get on a single T1. So, when we announced support for 32 T1s, it seemed like 'that's ridiculous.' No one will ever prove you right or wrong. Now, of course, we're up to three times that and other competitors are getting up there and Timeplex is talking about how they're going to support 144 T1s, although now it's a year and a half later than had been believed, but the issues of capacity are big ones because the economic imperative to go to T1, I happen to believe, took most everyone in this industry by surprise, and the number of T1s installed, the amount of traffic being put on them and so on and so forth, was just dramatic. There were several points where we thought our growth was simply going to be limited by how fast we could grow our manufacturing capabilities, and that to the extent our competitors would get business, it was because they had shorter lead times than we would at any point in time. So we saw a market that was like having a tiger by the tail and trying to keep up with it.

Pelkey: And this was in '85 and '86.

Forkish: No, I think that the issues had largely become more mature today. People are worrying about the types of issues that you referred to. I think that the next great leap forward will be in the -- the network of networks issue today is the wide area network interconnection with LANs, and that's going to be required to integrate applications, communications and distributed applications, which are now largely LAN based. That's going to be required to have the geographically disbursed offices communicate with each other and, from a network management point of view, it's going to be required because there are MIS organizations today all over this country that are worried about the proliferation, two years ago of PCs, today of LANs, things out of their control, but which need to be brought into the information infrastructure in order to tie the information together and create decision support systems and so on and so forth. You see it in industries that are very competitive and where information matters, so we do very well in the brokerage and financial services industries. They tend to be a little bit more competitive and leading edge these days, and I think enough ground will be broken that there'll just be a huge follow on. We used to think that our product would never be installed in smokestack America, but I'm beginning to doubt that. I think more and more that that's simply going to be the next wave.

Pelkey: If we might go back to that issue of you being a datacom company, coming then into the voice world, somebody said you overcame that by just sheer competence, that you spent time with people.

Forkish: We did have engineers with backgrounds, and most notably Walt Gill, who came from Avantek, had very much of a transmission background. Walt, and on the Cohesive side, Jerry Dooley, who came from AT&T, were one man armies for the two individual companies, happened to go out and lend this air of credibility. My own background was in networking software, and although over time I learned and became conversant with so many of the issues, it was pretty clear and people were concerned about the fact that I wasn't able to rattle off all the different types of analog interfaces and, you know, the different AT&T specs and so on and so forth. So that became a litmus test, and we had to get smart on that very quickly. Now, that's a problem for everyone and having a sales force that's smart on those issues, and we were certainly wearing out the airlines in those days going out and supporting the sales force just to get the basic business.

Pelkey: The T1 multiplexer, the T1 switch, at some level one could argue it was the first product in the datacom industry that started to encroach upon the traditional voice community, bringing in the concepts of computer control and digital and so on and so forth to something other than big central office switches which were still (unintelligible) oriented.

Forkish: The other thing it did was, we were a little bit easier to swallow from a large organization's point of view than some alternative technologies which probably were never viable in the first place, but back in the early '80s, there was an awful lot of talk about integrating voice and data in the office instead of outside of the office, so we allowed them to exist independently and only come together going on the external facilities so they didn't have to worry about having data get involved with PBX selection or putting phones on their Ethernet and so on and so forth, just all these horrible nightmares that both sides had about how they might have to interact with the other. It was so obvious from sales situations that you could sell your solution to the data side or you could sell it to the voice side, but it became compelling when they both worked together. When they both came together in a meeting and they were introducing themselves to each other, you said: "Ok, there's something very interesting going on here." We simply saw a lot of that, and the politics of large organizations was very much a factor in our selling and our selling strategies.

Pelkey: That must have been a very interesting experience to witness that, the user environment where the data and voice guys walking into a room for the first time, and just totally confused the world.

Forkish: Well I somewhat facetiously used to say that, when we were trying to recruit salespeople, they better have been camp counselors before, because they were going to need that experience.

Pelkey: How true. This process, then, of technological origins and intellectual property -- it was less -- not to minimize the achievement of IDNX, but you weren't necessarily creating new ground there. What you were doing was you were taking this know-how that had been generated over the prior five or ten years about communications, and having a team with different skills and, having seen clearly what this problem was, therefore having a clearly defined set of objectives and what you wanted to accomplish, and going about it, that was --

Forkish: The work that we did in networking was so dramatic in the T1 and circuit switch world and would have been viewed as so mundane by anyone with any experience in the packet world, was very ironic to me, particularly when you consider the number of dollars on the voice side, it seems that would have attracted that sort of expertise in the past. To some extent it did. You know, the signaling systems in the public network and so on and so forth are pretty sophisticated networks in and of themselves, but the people who were supplying products, the first generation T1 point to point multiplexers seemed to have a complete dearth of understanding or capability to build such systems, and we were aware of various kind of old line companies. We would hear 'Wow, they've just thrown 38 programmers on the problem to create software as good as IDNX,' and it would always make us feel a little more confident when we heard that, because that was simply impossible. You weren't going to solve the problem with a large number of people. You had to get experienced people who had solved these problems before, and it was a people management issue too. You really had to mix up and stir well people from different disciplines and hope that they were able to integrate themselves in order to create a product that integrated their disciplines.

Pelkey: There was this networking concept that was really bringing that to the voice world, which was really the blending of viewpoints, paradigms, that lead to the successful product. And networking, the issue of setting circuits up and having alternate routing and having fast alternate routing and collecting information so you knew what was happening and why it was happening that was -- bringing that perspective to the problem as well.

Forkish: We would describe things that, for example, in the network, you only needed to know the end points. You didn't have to plot the path through, and to many people, that was simply magic. I mean that was just the most astounding advance they had ever heard of. Again, in packet networks, you have all kinds of aliasing capabilities and closed users groups that what we had was pretty mundane in that regard. The fact that you could add a new node to the network without having to go reprogram the other nodes to tell them of its existence, that they discovered it and were able to use its resources immediately, was just -- people were flabbergasted. They had to see it demonstrated, in many cases, in order to believe it, but once they saw it, and then they looked at the amount of direct labor they had in trying to manage static routing cables and so on and so forth -- you know, that was network management then, and we were relieving them of the big network management burden.

Pelkey: And those concepts about configuration and so on, as you said, those were simplistic notions out in the --

Forkish: You would have been laughed out of a sales situation if you came in with a packet switch that didn't have those. IT was just -- you didn't even start talking about product capabilities until you had all those bases covered.

Pelkey: Right, even though it was a virtual circuit, you still needed to know it could reroute itself and get to where it needed to, because of its nature. I have no other questions. It's been very, very helpful. Thank you very much for your time.

Forkish: Thank you, I've enjoyed it.

END OF THE INTERVIEW