

An Interview with
NORMAN HARDY

OH 412

Conducted by Jeffrey R. Yost

on

4 April 2012

Computer Services Project

Portola Valley, California

Charles Babbage Institute
Center for the History of Information Technology
University of Minnesota, Minneapolis
Copyright, Charles Babbage Institute

Norman Hardy Interview

4 April 2012

Oral History 412

Abstract

Norman Hardy, one of the early and continuing technical leaders at Tymshare, Inc, discusses technological and organizational development of the company. Among the topics discussed are the adoption of new systems – SDS 940, DEC PDP-10, IBM 370 – and the development and evaluation of TYMNET. He also discusses early context to time-sharing, the long-term challenges of the proprietary network model in light of the internet and applications.

Yost: My name is Jeffrey Yost, from the Charles Babbage Institute at the University of Minnesota, and I'm here this morning on April 4th, 2012, with Norman Hardy for an interview about Tymshare. I'd like to begin with just some brief biographical questions; if you could tell me where you were born, where you grew up?

Hardy: Yes, I was born in southern California, 1933. My folks had just traveled out to California when my father had lost his job in the GE labs. And he got out here and found out by telegram that he wouldn't be able to; that they were laying off people so he stayed in California. So that's how I got to California. I went to school in Berkeley in 1953 through 1955, and graduated, and went to Livermore.

Yost: What did you study at Berkeley?

Hardy: Math and physics. I got a degree in math, and had quite a number of physics courses.

Yost: What were your career aspirations at that point?

Hardy: Computers were already on the horizon and I was very interested in computers. When I got into the math department at Berkeley I began running into people that were better mathematicians than I was, more and more often as I got up towards the graduate level. Computers were new and not many people were into them, and so I switched to my second love of computers. And Livermore was a beautiful combination of all three; math,

computers, and physics. And they had the first of any of the large computers, and so that was a perfect place for me to go. And so I did both physics and math there; and wrote programs to put them together.

Yost: Do you recall what model of computer you worked on in those days?

Hardy: Oh, well when I got there, the big computer that had been there only a few months was an IBM 701. And I didn't write a lot of code for the 701. I wrote some code for some of their smaller machines like the IBM 650. But when the IBM 704 got there, then I had been attached to a physics program on the 701, not as programmer but as mathematician, and on the 704 I began to write some code for that machine. And I wrote some novel mathematics for that machine.

Yost: Did you pretty much teach yourself to program, or did you have mentors that helped?

Hardy: I've recently been thinking about that and I realized when I got to Livermore there was a 19-page manual that told you everything you needed to know about the 701. Now the manual for the x86 that comes in my computer is 3000 pages, not counting systems software of which the 701 had nearly none. Nineteen pages contrasted with 3000. So, in a weekend, you could absorb those 19 pages and, in some sense, know as much about computer science as there was known at the time. Now, as a programmer, you weren't productive because it was a lot of work to write a little program but you were

on par with everyone else. Well, there was [a] one-page description of a; two-page description of a square root routine, and an assembler, which was described in a few pages. That was it. (Laughs.) And so there were a lot of programmer-physicists out there, at the time. Well, that was, I found, very productive.

Yost: How long were you at Livermore?

Hardy: I was there for about three years when the IBM Stretch was being negotiated. IBM and Livermore. And so about a year before the Stretch was to be delivered, I went to IBM and worked on the Stretch project, and contributed there a little bit to software and late engineering designs. Then I came back to Livermore and worked for another couple of years. I'm terrible on dates, so I have to go by relative spacing. Then something happened. I'll be chronological here. M.I.T. had got Project MAC underway, and there was a CTSS, Compatible Time-Sharing System, at M.I.T. and it had just become stable, so that people were using it and it was not crashing, most days. And so they invited a number of people from industry and government laboratories, and I was fortunate enough to go back and visit Project MAC and CTSS.

Yost: And did you meet with Robert Fano and Corbató?

Hardy: Yes. There was just a bunch of luck here. My officemate was Gene Amdahl. He had come back, also, to look at this. So that's where I got to know Gene. There were two important things there. I had gotten my first hands-on experience with the time-sharing

system. I had known the idea for several years, going back to John McCarthy's early papers, which you should look at. But this was my first experience with time-sharing system and the structure of a real live system that was easier to see. At the same time, there was a series of lectures on Multics, which was still several years from fruition. But Bob Graham, Corbató, and a couple of others were giving lectures to this group on where they were going. Both CTSS and Multics, as it was foreseen, had some wonderful ideas I drew on for the next 10 years. So M.I.T. is an important intellectual source for the beginnings. And before that, McCarthy at M.I.T. had written some of the seminal papers on time-sharing.

Yost: At that time, what were your expectations about the future of time-sharing?

Hardy: I was very much a technology nerd, it hadn't even occurred to me whether; I don't recall imagining an industry, a computer service industry, at that time. I suspect that jumping a few years ahead, that when I heard that Tom O'Rourke; well when I heard that GE had set up a time-sharing service, and mistakenly, I was not very impressed with that.

Yost: Was that influenced by the Dartmouth time-sharing system?

Hardy: Yes. I looked at the Dartmouth time-sharing system; I read papers about it and I was not impressed with it until I learned about the hardware that they'd done it on. Then I was extremely impressed because they had done this very useful system on piddling hardware. It would not have occurred to me that you could do something so useful on the

hardware that they did that; what was it? The 235 or GE; it was a little minicomputer. Kemeny did a fantastic job, in retrospect and I never saw that system up close.

Yost: When you were at Livermore did you know Tom O'Rourke?

Hardy: No. I met Tom just a week or so before I left Livermore. I'll get into that sequence but going back a little bit, subsequent to CTSS, the Project Mac experience, Livermore was taking delivery of a CDC 6600, eminently. And I talked Sid Fernback into designing a time-sharing system for the 6600 for Livermore. It's hard to say who gets more credit for that. I had a lot to do with the top level design, but there were some bright, opinionated people out there who did a number of things that I didn't always agree with, but the result came out actually very well.

Yost: Were those people at Livermore or CDC?

Hardy: Livermore. Bob Abbott, Cliff Plopper, Shigeru Tokubo, Fraser Bonnell, and a couple of more whose names I do not recall. The system was originally in assembler but large parts of it were rewritten in Livermore's extended Fortran. The system morphed into LTSS. Livermore was a strange place. I don't think it occurred to almost anyone at Livermore to export the software we did there out of the lab. It wasn't especially a matter of security, it just didn't occur to us. That time-sharing system for the 6600 did get used at Los Alamos so that there was some cross fertilization there.

*Revised 2014-01-10 to correct spelling of Bob Abbott and Shigeru Tokubo, p.7.

Yost: Using IBM systems, were you at all involved in IBM SHARE?

Hardy: I wasn't. SHARE was subsequent to Stretch; and subsequent to Stretch, IBM didn't have much of a presence at Livermore. They had no big machine subsequent to Stretch until well after I left. So I didn't have much to do with SHARE.

Yost: At the time that you were working on plans for the 6600 time-sharing system, were you aware of Project GENIE at that time.

Hardy: No, I didn't learn about Project GENIE; I learned about Project GENIE when I started talking to Tymshare; I guess they already had their name when I first heard of them. So when I first started talking to Tymshare; well, Tom O'Rourke and Dave Schmidt left GE planning to buy hardware and software from GE to go into competition with GE. And after they left, GE began to have second thoughts about selling to their competition.

Yost: Had GE already begun to sell time-sharing services by the time they left?

Hardy: Yes, that's my recollection. It was just very near the beginning of that. So, I heard from Tymshare about Project GENIE. I should have known about Project GENIE because I was all over Berkeley, but I hadn't run into them.

Yost: And how did you come to begin to work with Tymshare?

Hardy: Another IBM episode intervened. IBM began the ACS Project over here on Sandhill Road in — oh, I'm terrible about dates — must've been about 1966. Don't believe any of my dates; don't trust in any of my dates. It was to have been a supercomputer a hundred times as fast; it was to have been equipped with a clock cycle of 10 nanoseconds, and execute a few instructions per clock cycle; a superscalar. They didn't know that when I went over there, but that came to pass; that design came to pass. The sort of the original superscalar RISC architecture, which was not already built, but was specified well enough that subsequently IBM went back in and it materialized eventually in their Power PC architecture. So, at that time, they were setting up shop here in their first developmental organization in California and I knew a number of their people from my past experience with IBM. So I came over and joined the ACS project. I was there for about a year. Oh, and Ann, my wife at the time, who was working at Livermore, we both moved over here and she went to work at Tymshare a year before I did. And indeed, she got the; she was the one that took the 940; Project GENIE had taken an SDS 930, modified it to an architecture that was subsequently called the 940 by SDS, and she worked on it for nearly a year before I came. And there were quite a number of customers. I think there were probably at least two 940s by the time I got there. Now I was; you know, being in the same house, I knew what was going on. I knew all of the Tymshare people, but I was at IBM. I had great hopes of the ACS being a wonderful time-sharing system. It didn't come to pass. And so ACS was winding down when I came to Tymshare. Ann had; the system from Berkeley had to be retrofitted for disks and

*Revised 2014-01-10 to correct 10 nanosecond ACS clock cycle, p.9.

drums to swap on and she did that work, plus quite a number of changes necessary to make it commercial.

Yost: In the early days, I understand she and Van Vlear were really the only technical employees, is that correct?

Hardy: That's about right. In the operating system area. There was an important part of Tymshare software which I can't tell you much about, the compiler and the application area. I was aware of it at the time but that was being handled by competent people and I just didn't get very much involved with it. It became an important part of the history that I can't tell you about.

Yost: So from the very start, there were plans to have applications.

Hardy: Yes, some of which came from Berkeley. Butler Lampson did this nice little, he called it CAL. It was a duplication of JOSS, which you've probably heard of; Jules Schwartz.

Yost: At Rand.

Hardy: Yes, at Rand. And it was quite a faithful transcription of JOSS; a slight improvement but by and large, just a duplication of that wonderful system and we used that. It wasn't a big part of our business but it was a significant early contributor to our

business. And there was SuperBASIC and then some early spreadsheet-like systems. I have a friend, Doug Crockford, who has nothing to do with Tymshare. Crockford has written some books on JavaScript. And Crockford has recently taken it upon himself to try to find out some of that information, and so he naturally came to me; and I couldn't tell him very much. But then there's another; the fellow who was running the compiler and application sort of stuff — his name will come to me in a few minutes — and I suspect he is around and can tell you about quite a bit of that. Arden Scott. I believe he had much to do with the early Tymshare decision support packages.

Yost: And when you were at Livermore, did you work directly with LaRoy Tymes?

Hardy: No, I didn't. I knew LaRoy at the time; LaRoy was a computer operator. There's no end of interesting stories to tell about Livermore computer operators, which I won't go into now because it's too far off the track. But he was one of these bright kids that came in for; I forget how much college he'd had; but he was very bright and I won't tell you about his accomplishments at Livermore. They went far beyond being a computer operator. Well, that's a little bit; I'll get back to LaRoy.

Yost: What were your first impressions when — obviously, you heard quite a bit, probably, about Tymshare from Ann — but when you first arrived, meeting Dave Schmidt, and Tom O'Rourke?

*Revised 2014-01-10 to insert Arden Scott, p.11.

Hardy: Well, Tom O'Rourke was a businessman and, how shall I say (pause); an interesting fellow. To me, he was a businessman who was interested in time-sharing. And this was an unusual; how shall I say? I guess I will be blunt, I never have had a lot of respect for businessmen. (Laughs.) But he was pushing something that I thought was exciting; thought was a good idea. And Dave Schmidt was halfway in between. Certainly computer savvy, and seemed to have a fundamental talent for knowing what was feasible. He wasn't much of a programmer, but he knew bullshit when he saw it. And for the position he was in, that was vital. I didn't look at Berkeley; he looked at Berkeley and saw that it was a good system. And so he made that choice, which was strategic. And he made that choice when it was hopelessly slow because they didn't have anything better than magnetic tape to swap on. And so it was leap of faith to know where he could get to, and to make a reasonable guess. And he was right.

Yost: And only a couple users could use the system at a time?

Hardy: That's right, only a couple of users could do anything on that system. But he looked at the system and saw that it was sound; and that it was a step ahead of what GE had; and the machine could be scaled up with drums and disks to be a serviceable machine, which is essentially what Ann did the first year.

Yost: And when you joined Tymshare, what was your initial title, and what was your initial role?

Hardy: Well, let's see. I guess it was in charge of operating systems, of which only the 940 existed at the time. And Ann went on to other broader software areas. I forget the timing, but Tymshare began to acquire other companies and I was not very often directly connected with that except insofar as there were tasks integrating software together; their systems with ours.

Yost: Was Dial Data the first acquisition?

Hardy: Yes, I guess that was the first system, I think. And that was the 940 system.

Yost: And that acquisition was after you arrived?

Hardy: I'm sorry?

Yost: That acquisition was after you arrived at Tymshare?

Hardy: Yes.

Yost: And for chronology, do you know what month you joined?

Hardy: No, Ann might.

Yost: Was it late 1966 or early 1967?

Hardy: Oh, I'm pretty sure it was at least 1967.

Yost: Okay. And in talking with Ann yesterday, I learned that LaRoy Tymes, back in 1966, had proposed the concept of a network to Dave Schmidt and he wasn't sold on the idea at that time. When you arrived, was that something at all under discussion yet, within the company?

Hardy: No, I don't recall any discussions at TYMNET about a network until after I got to Tymshare. We opened up a Los Angeles; we had two or three 940s here; we opened up a Los Angeles computer center and it was already entirely evident that we were going to expand. Setting up enough data centers all around seemed like a pain and we began to talk about multiplexed technology. I forget what it was; I think it was the telephone company sold frequency division multiplexers, which were; we tried them out and they were not very good. They were not adequate. And even between here and Los Angeles. Packet switching hadn't been named yet. Time division multiplexing had been tried. Many computers were quite new and the idea of using a minicomputer to do time division multiplexing; well, you could buy a minicomputer for a smaller cost than the time division multiplexers. And again, Dave Schmidt foresaw the flexibility of that. You could daisy chain the multiplexers; you could do all sorts of tricks; and so without much planning ahead; now, LaRoy and I — and Dave Schmidt was always in on this — LaRoy and I were the idea producers and Dave Schmidt was the guy that sold it to Tom, essentially. And he added his ideas, too. So we went into this with the idea that time

division multiplexing was going to pay off in any case. We had wild ideas of where it might go, but they were fuzzy. There was no plan. There was just wild ideas about where you could go with a programmed switch.

Yost: What were some of those wild ideas?

Hardy: We certainly imagined being able to; like a telephone PSTN, the POTS [Plain Ordinary Telephone Service (Telco slang)]; where circuit switching; we certainly imagined that something like that could be done. We knew the way the telephone company did it wasn't suitable for teletype traffic and we didn't know what was, when we first began to program these machines. But programming them gave us a lot of insight. So we got this SPC-12, which was a strange little computer by a strange little company in Los Angeles. It was just exactly what we needed, except for being too small. And we almost didn't have to add any hardware at all to that to make it a time division multiplex.

Yost: And how long were those in use before switching to Varian 620s?

Hardy: Only less than a year; we quickly ran out of capacity on those. They were sufficient to prove the algorithms and to provide service for a while. There was a suit for a while because; I wasn't in on the negotiation on the delivery of the machine. There was some contingent deliveries and they worked out we'd pay for them; and so we actually had; we did pay that company because we did put them into service for a while and they

carried a real customer load; the SPC-12s. But they were doomed by their success. Because they worked, but weren't big enough, we had to go on to something bigger and Varian Data machines; the only competition at the time was an HP machine; a 16-bit HP machine that would've also sufficed. But Varian Data machines had some slight advantages so we went with those. And we had to do a little bit more hardware on those. They didn't have any; the SPC-12s had had static registers that you could put a value in and they would provide voltage levels on; with one wire per bit in the word and that was just exactly what we needed to drive the teletypes.

Yost: And can you discuss the programming side of the network? The work that you did and the work that LaRoy Tymes did on the network?

Hardy: At Livermore we had the CDC 6600 which included 10 little minicomputers. They weren't really minicomputers, but they were virtual minicomputers. And we had devoted one of those to handling the bit-by-bit teletype traffic. The teletypes were so slow that our program could handle one bit at a time, running around all of the teletypes. And that, indeed, had had a precursor in some military communication systems, where I believe the NSA had written some programs to run on early UNIVAC machines that would sample many teletype lines without the \$1000 per line cost of a serializer. I got that idea from someone back; I don't know where I got that; I visited NSA a number of times and I suspect I picked it up there. And so that's what we did to drive teletypes at Livermore, so we knew that software could drive teletypes with minimal hardware addition; which we did on the SPC-12s.

Okay, let's see. So LaRoy and I most of the early thinking on; okay, the next step, I believe, was statistical multiplexing. The first version of time division multiplexing took no advantage of the fact that most teletypes were idle most of the time. And which was a substantial factor. And so when we exploited that, we got a big advantage and there was; then there began to be variable amount of buffering in the machines, in the nodes. And so the characters came out at a; under a variable delay. AT&T had done something like this when they switched over to; they had had a network of these Baudot Code teletypes of five bits per character when ASCII came out and the Model 33 teletypes that were eight bits per character. And AT&T had had some kludges to interconnect such machines. So some of those ideas were in the air. An old telephone company precept was that absolutely constant delay between input and output. There might be some delay, but the idea that you could spread things like that in time was alien to most Bell System engineers so they very much resisted this idea of statistical multiplexing. I don't know; I'm just speculating that perhaps some of the business types saw this as a way to lose revenue. Any way you make their equipment more efficient, they lose revenue; which was a problem to strike Tymshare later.

So LaRoy began thinking of larger buffers. The machines were, even then, growing in capacity and you could afford to buffer characters and relay them through a path through the network. And we initially had fixed paths so that a given port on a machine; when computer manufacturers sold us machines, their plan was they would sell us a thousand dollar unit for each of the teletypes that we wanted to connect. And those were well;

those were discreet ports. We tended to remove that stuff or not buy it in the first place, and make connections between the TYMNET nodes — or just one cable between the TYMNET node and the host — and modify the software in the host and the network node to bypass this per-circuit hardware. But the concept of a host port remained, and one host port would go to one particular dial-up presence in some city, and typically in the closet of some time-share sales organization. You'd go into Denver and you'd find a one-, or two-, or three-room office and a small minicomputer there; and 10 to 20 or 30 dial-up ports there. So this fixed routing was in place for a year or so and it; even before we did that we realized that we'd have to move beyond it because that was too inflexible and it was too much work establishing, and changing, and coordinating this fixed routing.

Yost: Were you aware of the ARPANET project at that time?

Hardy: Yes.

Yost: Did that influence thinking at all?

Hardy: It certainly did. I can't tell you how, but these things were in the air and ARPANET was the biggest thing in the air. It was clear that ARPANET was forwarding packets and we were forwarding characters. And you can't afford to put all of the addressing overhead on every character. And so we came up with circuits. On a line between two nodes there would be little bursts of information for a particular user, and as little as one character, but often three or four or five. Or if he's driving a terminal up at

full speed, maybe 10 or 20 characters for that one user. And at the head of that, would just be an 8-bit identifier saying this is for user number 41. We had a lot more than 64 users, but 41 only had to distinguish the users that were on that particular link. So as a packet arrived at a node it would get disassembled into its constituent characters, put into the relevant buffer, and when it's time to go onto the next link it would get another number; it would be 69 on the next link. And so that was the biggest difference between TYMNET and ARPANET. And the ARPANET, the routing information that traveled with the data was constant as it traveled through the NET. Turns out that Cisco Systems used; all of these ideas were considered by everyone at the time to be open and free to the world. No one thought of patenting any of this stuff at the time. And Cisco subsequently picked up some other internal routing. My understanding is that when Cisco routers talked to Cisco routers they route the information with a TYMNET-like header; MLPS, I think; I forget what the acronyms are. It's still pure internet protocol on the outside but inside they use some of the schemes that can be more efficient.

Yost: Were you aware of any other competitors developing networks at this point?

Hardy: We were aware that there were some. But I was unaware of any of the technical details of them. I know that there was other statistical multiplex networks at the time. GE had one

Yost: UCC?

Hardy: Yes.

Yost: And Computer Sciences Corporation Infonet, was it?

Hardy: Yes, we did hear. Yes. But ARPANET was the only one whose technical details we were privy to. And AT&T would come talk to us. AT&T was good about trading ideas with us. We told them what we were doing and they told us what some of their ideas were.

Yost: Who were some of the earliest customers of Tymshare and did it fall in certain categories?

Hardy: Oh yes. One of our favorite customers because they gave us some good publicity was the National Library of Medicine. They came on when; well that's actually in the midlife of TYMNET. Some of the early customers; we were big in; we sold a lot to people like Lockheed. Not that Lockheed didn't have computers, but they didn't have time-share computers. Our computers were small, didn't have much capacity, but there were a surprising number of science and engineering jobs that didn't need much, excepting the turnaround was a huge strategic advantage.

Yost: Did you have a sense of what time-sharing capabilities meant to these aerospace and defense contractors that had big mainframes, run in batch, but could not do time-sharing?

Hardy: I know that SuperBASIC was suitable for what would've, in the days of the 704 — 1957, 1958 — would've been the small FORTRAN jobs. Already, the 1958 FORTRAN; the size of job that you could in 1958 FORTRAN would exceed anything that we could do eight years later in however many; so we were definitely limited to the smallest jobs. I came from Livermore, where we had done a time-sharing system on the 6600. I looked at the 940 and said this is a miniscule machine. What can you do there? I saw one application that convinced me that trivial little jobs can be highly strategic. It was a printing job; this guy had written a program for a small job shop printing operation. They would go out and bid on printing jobs and there would be 10 or 20 or 30 parameters to the job. How many colors? What was the size of the paper? What was the quality of the paper? What was the length of the press runs? Many of these little details, most of which were obvious but I would have never thought of, that go into making a bid of a print job. And overlooking one of these details can sink; can wipe out your entire profit. This guy wrote a program to go over this list of, the checklist, and make a bid. The whole program would run, maybe execute 100,000 instructions, which is microscopic. (Laughs.) And get them a \$10,000 contract. For the first time it occurred to me that really trivial computing can be strategic. I hadn't seen that before with my background at Livermore.

There were a lot of these small, really small companies that you'd never; you know, Mom and Pop shops that knew what a computer was for, had figured it out somehow.

Yost: When you were designing TYMNET was computer security; did that ever get discussed or come into play?

Hardy: Not in the modern sense. Our lines were unencrypted; it would not have resisted wiretapping; frankly, I never heard of anyone wiretapping a digital line back then except; of course, NSA was tapping into the telegraph, telegram business digitally, which came out in the *The Puzzle Palace*, Bamford's book [*The Puzzle Palace: A Report on America's Most Secret Agency*, 1983]. One, it was certainly not in the air. And if it was, I wasn't aware of it. And I was in a fairly good position to be aware of it. I'll put it this way; I believe that there weren't any amateur digital wire tappers at the time. I won't hazard a guess as to whether NSA was into the; whether the professionals got into it at that time.

Yost: And you weren't contracting with the DoD?

Hardy: I was not contacted by; there was no contact (pause)

Yost: Contracting. There were no DoD agencies that had classified documents that Tymshare was contracting with early on?

Hardy: Classified; NSA had a contract with Tymshare but it was a completely unclassified; Intel had come out with the 4004. Intel wrote a program that ran on our computers that emulated the 4004 so our customers could check out software that was

designed to run on the 4004 before Intel could deliver one. NSA was such a customer. But it was entirely unclassified.

Yost: Right. It was around the 1967 period, Willis Ware and others are starting to write about the impact time-sharing is going to have on the classified community and how it poses challenges.

Hardy: There is; if the government had wanted to set up something, such as they now seem to be setting up with common carriers, at that time, they would have had to come through someone like me, or there's one other person that could've come on; the guy who ran our computer centers, and whose name will come back to me in a few minutes. Bert Novak ran our computer centers, and he would've had to have been involved; or one of the two of us would have had to have been involved. And I asked Bert sometime a few years ago about that, and he said there had been no contact.

Now, about computer security, our computer security; unless you got inside our walls, it was pretty good. Already the hacker mentality of trying to poke security holes from one customer to the next had already developed at various time-sharing organizations and at the time, I believe that most of those operating systems were highly secure primarily because they were vastly simpler. The kernel was ran in; well, the whole machine, including the customers space, was 64k words. And the kernel was less than a third of that. So instead of having gigabytes of code you have kilobytes. You can get kilobytes of code right; you can't get gigabytes of code right.

Yost: So you were able to understand operating systems in a way that you simply can't today?

Hardy: Yes.

Yost: The end of the sixties was a period of rapid growth and a number of time-sharing companies were launched, I think between 1968 and 1969 there's more than 45 new time-sharing companies that entered the industry. How did that change Tymshare? How did the culture of the company change?

Hardy: Well, let's see. There were a number of other; time-sharing companies would specialize in market segments, in industry segments. They would buy a PDP-10 or a 940, or later, a VM370 system and one of the eminent patterns is that Tymshare or other companies, larger time-sharing servers, would buy them. So there's a fair amount of consolidation with careful attempt to conserve the customer base and the specialized technology to serve that base. Another strategy is, we came to the point of selling TYMNET services to our competitors because, well, for a variety of reasons. One of the arguments was that some of these people who needed TYMNET had software that wouldn't fit on our machines. So that was one of the fundamental reasons to sell TYMNET service to our competitors. And a number of; well, not a number of; we sold TYMNET services to several categories of companies. GM might want to distribute access to GM's computers to GM departments that were far flung.

Yost: You mentioned NIH.

Hardy: Yes.

Yost: There was a Poison Control Database?

Hardy: Yes, they had a medical database, which their charter was to make it available. And they took that very seriously and on-line availability was key. With their system, yes, we attached TYMNET to their computers in that case.

Yost: Was that one of the earliest instances . . .

Hardy: I believe it was.

Yost: . . . of connecting TYMNET to; that was early 1970s?

Hardy: Again, there is an archaeological; National Institute of Health, National Library of Medicine has a small website, or at least a number of pages devoted to the early TYMNET days.[<http://www.nlm.nih.gov/archive/20130423/bsd/historypresentation.html>]

Yost: Okay, terrific.

Hardy: And Stanford University also had a similar medical collection. That was subsequent to NLM, but we also attached TYMNET to their computers to get international access to their database. This was when ARPANET still had the authorized use policy and it was a political hassle to extend ARPANET to some of these areas even though no one would argue about the utility.

Yost: Are you aware, were any of your competitors in the time-sharing industry that had built networks, had any of them opened their network to connect to [interrupted]

Hardy: I am not aware that anyone else did. No.

Yost: My research to this point has shown Tymshare and TYMNET as unique in that regard, but wanted to see if you were aware of any others. What were the greatest challenges early on with TYMNET, both technical and business?

Hardy: Let's see; well, the TYMNET Supervisor was a challenge until fairly late in the game because we didn't have computers big enough. It strained; when a person would log in through TYMNET2 with this dynamic routing, a signal would go to the TYMNET Supervisor saying that you needed to go from point A to point B. And at this time, this would have several hundred nodes so A and B each would be some pair of nodes in this. And the Supervisor knew the topology; knew every case of one node being connected to another. And it turns out there's a fairly simple efficient way to find a path but that was beginning to strain the 940s, which was where the first Supervisor ran. And we had a

couple of not-very-successful attempts to move it to another machine. And I had left the project by the time; I can't give you a good story about how it was solved. For a while it ran on a Sun work station, which was much, much more powerful than a 940. And at that time, I think that was the solution. So we had a problem with the network Supervisor having enough power. And when a Supervisor would go down, there would be a few minutes when no one could log in while some other Supervisor managed to discover what was going on. I think that was the primary; all of the other problems like network outages, we had diverse routing. Diverse routing was a fairly natural outcome of our architecture. So I think the Supervisor was the biggest technical problem/challenge.

Yost: Do you have a sense of how much Tymshare was competing on price versus differentiated service and how that might have changed over time?

Hardy: The personal computer essentially did us in, did time-sharing in. For whatever reasons, our general pricing did not go down very much and the people in charge of pricing were pretty sure that we couldn't expand a lot by bringing the prices down. I was skeptical about that. I thought that we could have stayed in business quite a while by bringing prices down. I'll put it this way, I'm very much an armchair economist. I admit that there are many details of the pricing issue that I don't have a visceral grasp of, and the people who were in charge of it had the opposite impression. I know that we did special deals with special customers, not all of which were profitable for us but we didn't; we did not have a good cost basis for our service. We couldn't tell you very well how much it cost to support a given customer. We worked on that; I worked on that. I made

myself unpopular in parts of the company by saying hey, that some particular customer is costing us more than we're getting in revenue from them. But that was not a large part of our business. I think we could've done better if we'd have had lower prices later on in the game for small users. I knew people, personally, who said I can't use time-sharing, it's too expensive. And I knew what the costs were, and I knew we could've been serving them with a good profit. But the tendency within the industry, as I saw it, was to keep the prices high and just go out of business instead of trying to get into a new business at a lower price, despite competition. There was competition in time-sharing and I don't know why the competition didn't bring the prices down; which is what an economist would have told you.

Yost: Was the pricing fairly similar between the different competitors?

Hardy: When they were comparable, yes. Sometimes, one competitor would have; well, the main pattern was that one competitor would have a proprietary app, industry-specific package they would get a good premium for. And we did that, to some extent, although that wasn't a big part of our business.

Yost: Application packages?

Hardy: Application-specific software, yes.

Yost: And was that charged for on a separate basis rather than standard in a bundle?

Hardy: Yes, there would be a premium to use some of those packages. The more standard package like a spreadsheet or compiler there would be no extra charge for.

Yost: Were there programmers brought on to staff to build new packages or was that also done by acquiring software products?

Hardy: Both. Tymshare had a few developers. We did the SuperBASIC package and we contracted out. I wish I could give you the names of the products but I've forgotten them. There was a spreadsheet-like package that was done by another organization all on their own. They came to us and said we'd like to access to this via your system. We did that and it worked just fine. Many of our customers would write applications; not many. A few of our customers would write specialized applications and then realize they could sell them, sometimes with our help, and sometimes without our help, to other users. Sometimes we would act as sales agents for other people's software. One of the good things about Tymshare was it was very flexible. You walk in the door to a sales office; here's a plan for us to make some money. And our tendency was to say yes.

Yost: How much did that extend from the managerial perspective of Tom O'Rourke?

Hardy: Oh, I think that was Tom O'Rourke's insight. Any way to make money is fine.

Yost: You mentioned TYMNET2. To make sure I have my terminology right, what were the distinctions between TYMNET and TYMNET2?

Hardy: TYMNET2 was when you could build a circuit for a new session based on current network topology and traffic. When you could decide to go from point A to point B on a moment's notice instead of having to go through planning; and several weeks worth of planning to set up a circuit. It's a little bit like the difference between AT&T dialing another person and establishing a private circuit; a leased line. That was the division between TYMNET1 and TYMNET2. Mischa Schwartz wrote a book on networks. *Telecommunication Networks: Protocols, Modeling and Analysis*; he has quite a bit of technical detail.

Yost: Great, thanks. I came across something that said that it was 1979 that was the year that TYMNET became a wholly owned subsidiary.

Hardy: That's about right.

Yost: What was the rationale behind that and were FCC regulations the reason for that?

Hardy: There was a period where there was discussion of networks such as TYMNET being regulated. I was not very close to the details on that. If any politicking was going on, I was not plugged into that. I know that there were some mixed feelings within TYMNET; whether that would be good. Some people looked at AT&T and sort of were

envious of the guaranteed profit. Others were worried about the stagnation that would be a likely outcome. So, okay; there was another part of your question about '79; oh, a wholly-owned subsidiary. You're right. I hadn't recalled that, that there was; being a wholly-owned subsidiary was related to the possibility of regulation, it just occurs to me. I hadn't thought about that for many years. It was done in such a way that I don't recall that at the level I worked at, it made a lot of difference. Now, there came a time when I got out of the; TYMNET software; I had a hold of TYMNET software but not all of TYMNET, by any means. Later on, LaRoy took over TYMNET with possibly the exception of hardware. There's another element of TYMNET that you should mention at least briefly. There was a hardware component to TYMNET that went well beyond the original Varian Data machines. Howard Steadman, who's a name you've probably heard, did some of the early and some of the later TYMNET hardware development. There was a couple of generations of machines that we haven't talked about there, that he contributed to.

Yost: Do you recall what those systems were?

Hardy: Yes, he was more of a hardware person and there was a very effective communication; the politics was good but the technology was; or the politics was bad but the technology was good, somehow, despite the politics. We came to good interface between the hardware and the software, and he would build specialized hardware, which was unlike anything the rest of the industry was building. Sometime, we need to talk

about the PDP-10s and the 370s, later developments; where he also did some development; strategic development for us.

Yost: Do you want to take a quick break?

Hardy: Yes, I was beginning to think that.

Yost: Before we took our short break you mentioned subsequent waves of hardware with the network. Can you talk about the PDP-10s?

Hardy: When we got into the business with the 940, DEC would have been another option. DEC, as I recall at the time, was one of those few companies that really understood the promise of time-sharing; more than XDS — SDS became XDS so pardon the confusion — what XDS did well is faithfully copy the hardware modifications that Project GENIE had made to the 930. They copied it well, they documented it well, and they delivered the machines. But they didn't really understand what time-sharing was. DEC understood what time-sharing was. They were close to M.I.T. where it sprung up. DEC and M.I.T. were really close to each other. DEC had sold a PDP-10 to Bolt Beranek and Newman and both of those organizations, DEC and BBN, built early time-sharing systems. Now, let's see; this was subsequent to Project MAC, prior to Multics; and subsequent, I guess, to Dartmouth. But they; on top of the PDP-10 those two organizations built credible time-sharing systems that we could've gotten started with.

*Revised to correct spelling of PDP-10, p.32.

But those were more expensive machines and so maybe we couldn't have gotten started with them, just from a financial, money flow standpoint. Both of those organizations continued to develop their respective systems for the PDP-10 and so; let me bring in another strand here. In the early days, XDS had lent us a lot of money. We were in hock to XDS at the time, and they were pressuring us to move to the Sigma 7. The Sigma 7 was not an especially appropriate machine for time-sharing, for a number of reasons. It was to have been a competitor to the PDP-10 but they did not understand as well as DEC what time-sharing was about and their architecture wasn't suitable. The good news for Tymshare is that the 940s had produced a good cash flow and we were out from under the pressure from XDS to use the Sigma 7. And we got –

Yost: Do you recall roughly when that was?

Hardy: That was in 1970.. Tymshare was in an acquisition mode. There was a company in Seattle called Computer Center Corporation that had a PDP-10. So we acquired our first PDP-10 by acquiring a company that had a PDP-10.

Yost: Was that a breakthrough to acquire a company that wasn't using the same SDS hardware?

Hardy: I think it was, yes. Now, we had bought some companies that were not in the computer service (pause)

*Revised to correct names Computer Center Corporation, p.33.

Yost: Not in time-sharing services?

Hardy: Yes, the time-sharing service business.

Yost: Do you recall any examples of those?

Hardy: I forget the order. Let's see; one funny little company we bought was a company whose business it was, when someone had a fire in a computer center they would call up their insurers, their insurers would call up this company, and this company; this guy; one or two or three people in this company would go out and look through the fire; walk through the place and say I'll pay you \$10,000 for that, \$2,000 for this, \$10,000 for that; and he just knew how to evaluate fire damaged equipment because he would take it and fix it. We bought that company. I had no idea why we bought it but it turns out to have been a good deal. Just little things like that; Tom O'Rourke says hey, if it's a good deal, we'll do it.

Yost: Did Tymshare get into custom programming services? That business?

Hardy: Custom? I'm sorry?

Yost: Custom programming services?

Hardy: Not very much. Occasionally; now, some of our sales people — a few of them — were quite competent programmers. And they would go and do marvelous things for their customers, and we at Cupertino would never even learn about it; which is fine. We would never know about the field sales people.

Yost: It could've been bundled in to get some business, but it wasn't established as a strategic business at the time?

Hardy: Sometimes, occasionally, it would be bought in; sometime in the field, some Tymshare person in the field would write a package that was good enough that it would get moved up and spread around the other sales divisions. And we encouraged that.

Yost: Can you talk about how communication worked so that you would get feedback from customers? Through sales, was that; and with larger customers were technical staff going out with sales staff?

Hardy: From my perspective, that was always rather haphazard. I was always complaining that I wanted to talk to more customers. Occasionally some critical customer would have a critical question that only I could answer because it was in my area. And that was fun. But that was quite rare; that was only a few times a year; for, you know, couple, three days a year something like that might happen. And I think that was the case with our other centralized programming staff, that I felt that we were too much isolated from the customers. I don't know what the hesitancy was to get us involved.

Yost: You mentioned that after a while, you weren't as directly involved in TYMNET.

Can you go through what job titles you had and what managerial roles you had?

Hardy: When the 370 came along, I got more specifically involved in that. And also, toward the end, this new project, which we haven't mentioned yet for a capability-based operating system for the 370, which is a saga. We could spend hours; that project substantially outlived Tymshare but it didn't come to anything financially significant. That was a project to go off and solve a problem such as I referred to in Derwent. There were several customers that had come to us with requests that; to establish a data marketplace and a software marketplace where there were security barriers to be put in place that classic operating systems could not and cannot solve.

Yost: And that discussion that we had, where you told me about Derwent, that was off the recording, so if you could go through it again.

Hardy: Okay. Derwent is an English company that had, and has — I believe they're in the same business now as they were then; I recently went back to look at their website; best I can tell they are in the same business — they have a proprietary encoding of chemical patents. I don't know how many of those there are, but it's a substantial database and they have a proprietary way of formatting them to make it convenient to do a variety of searches that I don't know enough chemistry to describe. The plan that they described to us; what they wanted to do was to allow their customers, who were typically

researchers, to search their database for, I presume, molecules that resembled something that they were researching. Indeed, in some cases, Derwent's customers would need to write programs to examine this proprietary data, run those programs in an environment where the proprietary data could not be copied back to the customer excepting in small chunks of information whose size was metered. So it's like I can send a robot to read your data; the robot can never return to me but it can send small packages back to me; and how much I pay Derwent depends on the size of those packages. Yes, I can copy the whole database back, but at that time I've paid Derwent for the entire development cost; plus profit, so that suits them. That was their idea. I thought it was very clever and we realized that we couldn't do it with our current software but that software like that could be written. And KeyKOS was the outcome of that. Tymshare and another company, Key Logic, did not succeed in making that commercial. It would've been a high security system with novel features.

Yost: How long, roughly, did you work on that?

Hardy: Well, that went on into the 1980s; through much of the 1980s before; Key Logic had some customers and provided some service, but it did not; the lower levels of the software were well developed and very robust, but we did not find a market for it. Most people were firmly of the opinion that someone would come along and add a module to UNIX and solve all of these problems. And that was not our insight; or that has not been the experience as none of them has added a module to UNIX to solve these problems.

Yost: When Tymshare went public, did that have significant impact on how the company was run?

Hardy: I don't recall that it had. I certainly recall the company gradually getting larger. I don't recall; of course, I'm probably the wrong person to ask. At a technical level you don't see that so much. Again, Ann would have a better grasp on that.

Yost: Once TYMNET, Inc. became a wholly-owned subsidiary, Tymshare was, of course, still using it; how did that pricing go and was there preferential pricing to Tymshare?

Hardy: That's a fascinating question. Let's see if I can answer that. I was worried about; I remember being concerned about the separation and that problems would arise, but I can also recall a year or so later that no problems had arisen. I don't know what the contracts were between those two organizations. Tymshare continued to be — at least while I was connected with it — the technical source of expertise for TYMNET. Now, later on, that political structure changed and LaRoy Tymes would be able to tell you more about that than I. You should talk to LaRoy; he's in Texas now — no, I think he's in Colorado now — but I've talked to him rather recently.

Yost: Yes. Ann made that suggestion as well.

Hardy: He could give you a lot of insight on that because he was connected with TYMNET more recently, substantially more recently, than I was. One other point that we should mention, but I don't want to break (pause)

Yost: Go ahead.

Hardy: The origins of VM/370 are; somewhere in your book you need to mention that that was a product of an IBM group in Boston that was close to MIT. It is now called the "IBM Watson Research Center in Cambridge". I don't think it was a research center — it was more than a sales office, it was less than a research center, in Cambridge. And so those people knew; they were close to M.I.T. and time-sharing. IBM, for reasons which are sort of interesting and I'm not sure I can tell you very well, came out with Model 67. It was a version of the 360 that had paging. Their plan was to produce a system they called TSS. They did produce that system but it was not very successful. The people at Cambridge; IBM at Cambridge, looked at the Model 67 and best I can tell, invented out of whole cloth, the virtual machine and they wrote a virtual monitor. It was called CP67. They wrote a minimalistic system called CMS that would run on a real machine or a virtual machine, and would serve one user at a time. They turned that into a damn good time-sharing system. When the 370 came out it did not originally have paging, but a year or two later IBM added virtual memory and IBM had seen the value in CP67 and produced VM-370. Again, they took the best ideas out of CP67 and replicated them in VM-370. That's the origin of another significant thread. That was IBM's successful

entrée into time-sharing. That's how we got; how Tymshare got their start in 370 time-sharing, too.

Yost: Now, the Service Bureau Corporation, a subsidiary of IBM was in the time-sharing business, wasn't it?

Hardy: Oh, yes, it was.

Yost: When that was sold in a settlement (pause)

Hardy: That's right! They were sold off in the settlement of a patent suit or something.

Yost: Yes, to Control Data Corporation.

Hardy: Yes.

Yost: And from what I understand, after that occurred, for a time at least, Control Data Corporation, was the largest time-sharing corporation in the country. Can you speak about CDC as competitor?

Hardy: Oh boy. There was some reason why they; I can recall them as a competitor. They were so far off in a different direction, in a different niche, that we seldom ran into each other. I wish I could explain that now.

Yost: Had you run into Service Bureau Corporation as a competitor before that?

Hardy: Yes. But they had such a different way of doing business that we didn't much run into each other. I'm not saying that they didn't have defined businesses; they were almost a different world. At this point, I cannot explain that but I have to come back and tie that in with another thread. Remember I was talking about the computer operators at Livermore. One of them was Ted Ross, which he and his son; well, just real quick. Ted Ross was a; no, he was not a computer operator, he was one of IBM's maintenance engineers. He would fix the machines at 2:00 a.m. when they broke. He was a remarkable person. Let me see if I can skip over some of the unnecessary detail. He was working for SBC when SBC was sold off to CDC. You know, Ted Ross is almost 10 years older than I but I'm not sure that he couldn't tell you some fascinating stories from his perspective. And Ted Ross is a remarkable person for reasons that I could go on for an hour or so about. So when you said SBC, suddenly that tied in. His son; I'll try to keep this down to 30, 60 seconds; his son wrote, for the X86, an emulator of the 370 and started to go into business seeing how we could commercialize it. IBM called him up and said we understand that you have a simulator of the 370. We have a deal; we would like to come out; here's the proposal. We'll bring out Carl Ross — the guy that runs this, the son that started a small company — we'll come out and bring out the crown jewels, which are how we test the 370s for compliance. The deal is whenever we find a discrepancy, we'll tell you. In the process, we'll learn about what your program is just by the nature of the interaction. So they did this; they found a few bugs; and a few months later, IBM bought

the simulator that they'd built. So, it's an interesting story. And Ted Ross had a lot to do with that simulator.

Yost: Interesting. What did the 370 system mean for TYMNET?

Hardy: Well, the VM-370 certainly was not the run-of-the-mill; certainly was a new direction. For the first time, we could talk to people in the data centers of the big corporations because we had something that was compatible. In some sense, it was competitive with their data center and in some sense, complementary to it. And so we did a; most of these large organizations didn't want to have anything to do with VM but we could run their PL1 jobs and other arbitrary IBM software on computers interactively. If they wanted IBM to help them solve that problem they had to buy a whole new VM computer and most of them didn't want to do that and so they let their people use our computers to do that. So that was a new business for us. And also, the 370s were bigger machines than we had so we could solve larger, more complex types of problems. We would bring people sometimes into the IBM world just because using our IBM machines gave them more capacity, in any one of several respects, than they would otherwise have. We did a fair amount of technical development in order to attach TYMNET to VM. If these companies brought in their own; if GM would buy a computer from IBM, a VM computer, VM-370, then it would not be attached to TYMNET and someone in Atlantic City from GM wouldn't be able to attach to it except through IBM's much more expensive and awkward communications scheme. And so that was another way we had our foot in the door. Hey, if you do your GM stuff on our computer then your people in

Atlantic City will have cheap access to it. So there was a number of tie-ins of that sort that were strategic.

Yost: I understand that in the mid-1970s, I believe, Tymshare acquired United Data Center.

Hardy: Yes.

Yost: And with that acquisition, Bernie Goldstein, for a time, was an employee of Tymshare and really became a force in mergers and acquisitions. Did you know Bernie very well? And can you talk about what he did at Tymshare?

Hardy: I certainly remember him. I'm afraid I can't tell you much about the business because I was not close enough to that. I'm trying to think who could tell you that. Bert Novak would know some of that. Oh, yes; now that; that pretty much washed over my head at the time.

Yost: Okay. Another acquisition — I think this was even earlier — was SRI's Augmentation Research Center; and with it its leader Doug Engelbart.

Hardy: Oh, yes; Doug Engelbart. Yes, yes.

Yost: Can you talk about the role Doug had with Tymshare?

Hardy: Yes. I don't know what was happening at SRI. I guess it was moving out of the research aura of Englebart's project and into a perceived production era. And personally, that was very important for me because we were just starting on the KeyKOS project.

And we just had (pause)

Yost: Can you spell that?

Hardy: K-e-y-K-O-S.

Yost: Okay, that's what I thought.

Hardy: Some people capitalize the K-O-S, but I don't. So a whole group at SRI, probably roughly 10 people, came to Tymshare. TYMNET was strategic for them because they needed to be geographically distributed. It was actually more strategic to some of Englebart's customers. The Air Force, in particular, was an enthusiastic user of that system. Their system had originated on the 940 and had graduated to the PDP-10. So that was just fine for us. We had PDP-10s and we liked PDP-10s; and there was a lot of technological confluence there. For my particular project, KeyKOS, we had just gotten the documentation religion, which was; and they had the perfect documentation system. I had never seen a better system than that, then or subsequently, for organizing data that needs to be shared and continuously updated. It was renamed Augment at about the time that we acquired it. And so we had ample access to that, which was wonderful. We got

that connected with TYMNET and the Air Force. We had a couple of PDP-10s I think devoted to that; I don't know quite what that configuration was. I don't know exactly what became of that project; the Air Force evidently lost some of their funding and that began to wind down later on. I don't know who could tell you. I talked to Doug Engelbart a few months ago and I think he's losing memory, so; I'm afraid.

Yost: What type of position did he have? Did he operate more as kind of a research scientist?

Hardy: Yes, very much a research sort of person. Certainly, guidance of; I don't think that; and with that (pause)

Yost: Was that common? Were there a number of others?

Hardy: No, there weren't. Come to think of it, that was an unusual; I had sort of moved over into a loose; well, I had a particular project I was tied into, but I was welcome most places in Tymshare if I wanted to stick my nose in. And, of course, Doug was already famous at that time, so he was more than welcome to wander around Tymshare.

Yost: Right. He's, of course, best known for advances in graphics. Did that have an influence on Tymshare in developing applications that used advance graphics?

Hardy: Englebart's system, while at SRI, did have some ability to show line graphics which had been crafted by the NLS author. This technology did not survive the trip to the Tymnet world. The following did survive: Englebart's Augment system provided 'WYSIWYG' on a modified ASCII terminal. This was direct interaction with a logically organized document with the fabled mouse. WYSIWYG of subsequent fame was in the production of a printed page where the author was in control of detailed physical layout, rather than a tree structured document. Augment could show the logical tree in many different views, and edit the tree thru any of these views. The reader of the Augment document was in charge of the display format, which might depend on the reason for consulting the document. I know of no such flexible system today.

The thing that he was the most passionate about was not especially; the graphic user interface, yes, that was an essential part of his passion but it wasn't the biggest part. The main thing that he sought to do was to develop the underlying information structures so that they would be comprehensible, modifiable, share-able. Well, his buzzwords were "intellect augmentation." How does a person work with the computer concerning ideas? To capture the ideas; to spread the ideas. To user interface was, of course, a vital part of that. But it was sort of only the surface.

Yost: That's like Licklider's mission; of course, Licklider and IPTO Director Successors funded him extensively with DARPA funds.

Hardy: Yes. Air Force contracts and, I believe, DARPA were supporting that and buying the service. I don't recall that there was any significant attempt to sell that service to

customers. And that, I think, was a strategic error because I'm a huge enthusiast of Doug's technology.

Yost: Are there areas I haven't asked questions about that you feel are important to understanding the development of both technology at Tymshare and TYMNET, and management of Tymshare?

Hardy: I'm sorry, I think I missed your question?

Yost: Oh. Are there areas that I haven't asked questions about that are important to understanding the development of Tymshare and TYMNET technology, or Tymshare and TYMNET as businesses?

Hardy: One pops to mind. I'll see if I can describe it in a minute. Someone asked me, a year or two ago, why internet took off and TYMNET didn't. And I hadn't thought about that much, and I thought about it for a while; and I came to the following conclusion. TYMNET was proprietary; not in the sense that we wouldn't tell people how it worked. It's just that we assumed, and everyone else assumed, that we were the only ones that would do anything to TYMNET. And people with a bright idea that weren't in the company had no way to contribute to its success. Somehow, ARPANET and internet developed in another way and anyone who had a little bit of money and bright idea could make money by building on it. You couldn't do that with TYMNET. It was proprietary not in the sense that we were keeping vital secrets. Now we wouldn't have shipped the

network supervisor but we'd love to have lunch with you and tell you in gory detail how it worked. (Laughs.) So we weren't proprietary in that sense, but there was just a culture around it that only we could add to it. And that lost out to the internet.

Yost: You mention that with the personal computer pricing was held pretty firm and it meant declining business in the first half of the 1980s.

Hardy: Yes.

Yost: Can you discuss the takeover by McDonnell Douglas, and what were they looking for, and what did they do with what they acquired?

Hardy: I was about to leave the company about that time. Well, in fact, it was connected. After MDC took over, my; here is a comment that I remember making at the time. The people from MDC that came to see what they had bought. They were some of the most conservative but intelligent people that I had run across anywhere, conservative technologically. Generally when I say conservative technologically I mean that in a negative sense but I'd have conversations with these people and they were really bright! They understood what they were doing, what we were doing, where we were proposing to go, and they didn't want to do it. It's not that they didn't understand, they just didn't want to do it. And so that's when my group spun out of the acquisition. I can only presume that their plan was to ride out and get the tail end of the time-sharing business. Oh, they knew it was going downhill, there was no doubt about that. They didn't know

where TYMNET was going. And indeed, my recollection — and I'm not good at numbers — but my recollection is that a few years later they sold TYMNET off for what they had originally paid for the combination of Tymshare and TYMNET; \$300 million; I don't know; in that ballpark. So not a great deal; maintenance but not development, I believe, was what we saw — what I saw — coming under MDC.

Yost: Tymshare and TYMNET got into helping businesses develop their own intranets. How big a part of that was the overall business in the late 1970s and early 1980s?

Hardy: I'm sorry, moving towards internet, or . . . ?

Yost: Intranets.

Hardy: Oh, *intranet*.

Yost: So, for corporations; helping them with their internal networking.

Hardy: We did sell; TYMNET cloned their system in a custom sense, to any one of maybe a half a dozen organizations. We sold a small TYMNET to DEC, which was interconnected with their network; and we sold them some system. I believe we may have sold; I believe we sold something to Canada for their national network. Now, I don't recall what that turned into. I know that TYMNET had nodes up there and I couldn't keep track of just (pause)

Yost: And was TRW another one?

Hardy: Yes, there was a TRW; thank you, I would not have remembered that.

Yost: Do you recall anything about that?

Hardy: No. That was smooth enough that the technical people didn't get involved, as I recall. There were talks with other foreign countries but I don't think that came to much.

Yost: Speaking of foreign countries, I know very early on, even before TYMNET, there was the establishment of an office in Paris.

Hardy: Yes.

Yost: How much of Tymshare's business was international? How did that evolve?

Hardy: Well, the original Paris time-sharing office was very much like the original sort of outpost in Los Angeles. We almost; now let me see; it was running for a while before we got a trans-Atlantic link. We got it, I think, probably about year; I think we tried a satellite link for a while and I think there were latency problems there. I think that was the first foray into satellite links. And then we got an ocean link, which was much more satisfactory. And, of course, then we'd approach; the question began to arise in

everyone's head, do we really need a computer in Paris? Because people were already gravitating toward the PDP-10s and the 370s, which; or at least I think we had the PDP-10s up at that time. We also had set up an office in London. Paris was first and then we had a London office, which was quite successful. So we eventually took the 940; well, the 940 was one of the first 940s to go but that was about the time we started contracting on the number of 940s.

Yost: I asked you about SBC and Control Data as competitors but in the early years, the largest time-sharing company was GEIS [General Electric Information Services] and I haven't asked you about them.

Hardy: Yes.

Yost: Can you characterize that business and what competition it provided?

Hardy: Well, GEIS was certainly competitor; I noticed them more than SBC; *we* noticed them. Again, these are my reports when I sit around a dinner table with some salesmen is the only route that I get such information. GEIS was, I think, more nearly in our style of; in our niche, so we competed with them a little bit more. Our particular technology was incompatible so that there was a cost going either way if a customer wanted to move. But I don't remember the marketing and the pricing struggles between those two.

Yost: Speaking of that, was there — when you got a new customer — was it fairly easy to maintain that business for a long time because of the degree of lock-in?

Hardy: Generally, there was a fair amount of lock-in. Even the various 940 systems that were out in the world would have incompatibilities. When we would buy a 940 company there would be a fair amount of work to do because we each improved the system and trying to get the important common elements into one system generally turned out to be possible. But yes, a customer, even moving from one 940 system to another, would incur some cost. There were certain kinds of things that could be moved over easily. But many of our customers would have been using the spreadsheets that; now I believe that some of these spreadsheets were owned by companies that were independent. There was beginning to be stratification there, where a company would own a 940 middleware, like a compiler, that they would have moved to two or three of the different 940 systems. But that was only; that had not developed much. Most of the compilers in the spreadsheet-like things were proprietary to a given service provider.

Yost: I think I've made it through most of my questions. Are there any areas that you can think of that I've missed?

Hardy: I can't think of anything, no. Let me see. There were a number of later unsuccessful acquisitions that we did, but that's probably not of as much interest.

Yost: Actually, if you can recall any specific names?

Hardy: Well, we bought; perhaps the biggest one was; it was a name, when we bought it you would have known the name; it was Unitax, I believe. It was; at the time we bought it, it took a mainframe to run tax preparation software and their business plan was to have people use their software remotely. And now I can't even remember what 'remote' meant. I think it meant something newer than U.S. Mail, but not much. They would send it in; do batch processing; and send it back to the people in the field. And so that seemed like a natural acquisition because with TYMNET you could make it much better. But now I'm talking really second hand, maybe third hand, so take it with a big grain of salt. There was also a management change. We left them pretty much intact in Kansas City, I think they were. There was a management change there that we should have done something about and my version, perhaps badly garbled, was that there was someone in management there that had a theory of how elements within their organization should communicate; namely, through channels. So they established the channels and were effective at cutting out the other channels. And the next year the tax software was very late, and tax software cannot afford to be late. And after it; and two years later the company was gone. So that was a failure. There was one other I had in mind, but I can't think of it right now. I think we got a little bit too far out of our league. Tom O'Rourke was willing to go almost anywhere but sometimes he went too far afield. Not often.

Yost: And what did you do post-Tymshare?

Hardy: Oh, I left just after MDC got there because they didn't want to have anything to do with KeyKOS and so we; Ann and I both went off and started Key Logic, which lasted a number of years and where most of the real development of KeyKOS happened. And Tom O'Rourke invested some of his money in it, and Gene Amdahl, and then a few other people that you probably haven't heard of.

Yost: Well thank you very much.

Hardy: Thank you; I've seldom enjoyed a morning so much.

Yost: Thank you. This is wonderful and I got a lot of valuable information for my research.