Oral History Interview with

William Franta, Ph.D.

On July 9, 2020
Minneapolis, Minnesota

Conducted by Jeffrey R. Yost, Ph.D.
Charles Babbage Institute

Abstract

This interview was conducted by CBI for CS&E in conjunction with the 50th Anniversary of the University of Minnesota Computer Science Department (now Computer Science and Engineering, CS&E). Professor Franta briefly discusses early education and interests. He attended the University of Minnesota (UMN), went on to complete his doctorate at the UMN in Mathematics/Computer Science orientation and signed on to be one of the founding members of the Computer Science Department as well as provided important leadership to the computer center as its Associate Director. The interview is especially rich in discussing the computer center, research projects with local computer companies, and the importance of the department to the industry and the industry to the department. Professor Franta did very extensive consulting with computer companies local and national while at the UMN (Control Data, Honeywell, Sperry Univac, Network Systems, 3M, and others)—along with government (NSF especially) important to setting up UMN CS labs. It is also offers considerable and important discussion of the 11 early faculty members of the Computer Science Department, their research and specialization and how the department was building coverage to educate from very early on after the department formation in artificial intelligence, programming languages, operations research, numerical analysis/mathematics, automata theory, and other areas. In addition to discussing his work at the University of Minnesota, he also touches on his second career in industry as senior executive at Network Systems and other firms.

Keywords:

Computer networking, Minnesota computer industry, Mathematics Department, numerical analysis, University of Minnesota, Computer Science Department, UMN, computer curriculum.

Yost: This is an oral history with William Franta on July 9, 2020 and it is part of the Computer Science and Engineering Department's history project for the 50th anniversary. Do you prefer William or Bill?

Franta: It doesn't matter. Whatever you like.

Yost: Ok.

Franta: I've been called both and worse.

Yost: Ok. Bill, can you begin by just giving me a little bit of biographical information, when and where you were born?

Franta: I was born in St. Paul, Minnesota, right about where we are.

Yost: Did you grow up in the Twin Cities as well?

Franta: I did indeed. When I was about to go to high school, there were choices, you could go to two, so our neighborhood kind of split. I went to a high school called St. Paul Murray. It's behind the fairgrounds, if you know where that is, on the west side of the fair grounds. And my siblings went to Washington, which is east of where we lived. We literally went to different schools together, as the old joke goes. But yes, I went to high school in St. Paul. To carry on a little bit, when I graduated, I'll always remember, the counselor took me aside, and I think I was number three in a class of several hundred at the time, he took me aside and said I should investigate a trade because I would never make it through college. And it's always troubled me since, that a long time ago, what did he really mean. I think what he probably meant was that I never did anything to study. I was building engines for cars and things like that. Maybe he said it from that perspective. Anyway, I didn't take up a trade.

Yost: Can you tilt your camera a little bit so it's a lower on you?

Franta: I don't know how to do that. Ah, there. Is that better?

Yost: Yes, that's perfect.

Franta: There! I guess I can bend the whole thing. I didn't realize that.

Yost: Yes.

Franta: Thank you.

Yost: Sure. Were there some subjects you liked more than others, pre-college, in school?

Franta: I was editor or one of the editors of the high school newspaper and I was very intrigued by language. Always have been. The well-written word, in particular. So, I was interested in language. Also there was a student named John Wolf, and I've reestablished contact with him. He lives in Florida now. And he and I, in the math classes, with a teacher named Miss Barschelet (SP?), we both remember her name, were allowed to sit in the front of the class and help solve the problems. So, this was probably algebra, maybe trigonometry, not quite sure, and we did that for a whole year. It was both a little nerve-wracking but retrospectively, it was kind of fascinating as well. The bottom line of that story is that I was reasonably good at mathematics, even in high school, although I didn't carry that thought forward to do the right thing initially in college.

Yost: And you did your undergraduate studies at the University of Minnesota, correct?

Franta: I did. When I graduated from high school—go ahead. Sorry.

Yost: What year did you start?

Franta: I started in fall of 1960. Long story if you want to hear it. I had two scholarships, to two different institutions. One was Hamline University and one was the University of Minnesota. I decided to attend the University of Minnesota. And I do not remember, each scholarship bore a name, — I'm sorry I can't find the names and I can't remember them. But the first year or two were under scholarship. I enrolled in the Aeronautical Engineering program, and quite frankly, I wasn't doing very well. I think it was because the mathematics they were using was in advance of the mathematics that we had been taught, and I was fumbling around a little bit. Although I didn't have the same trouble in Physics, so it's a little bit of an anomaly. Anyway, I was doing Aeronautical Engineering, and I was basically getting a couple Bs, Cs and once in a while a D. And let's see, the end of my sophomore year, the state of Alaska came to the Civil Engineering Department and asked about students who wanted to go to Alaska for the summer and do surveying. And I'm not quite sure how I finagled this being in Aeronautical Engineering, but I did. I spent the summer of 1962 in Alaska. We traveled a lot. Did a lot of things. I have a lot of stories, and we did very little surveying because we were on a roving crew, four guys on a roving crew. When I came back that summer—I mis-spoke, actually. I transferred from Aeronautical Engineering to Mathematics for my sophomore year. When I came back that summer from Alaska in '62, there were five of us that were majoring in mathematics that year, a huge number. It was literally—sounds like I'm being facetious, but it was a huge number for them. They apparently hadn't had that many undergraduate math majors in a while. In those days you could get a degree in mathematics, my diploma literally says mathematics, probably would say Bachelor of Science now. When I came back that fall, one of the other four members of the group said that the Computer Center was hiring students to operate the computer. We'll come back to this, but that's where I think my career really began, when I—so this is the end—the fall of my junior year in college, and I'm majoring in mathematics at the U of M.

Yost: Can you describe the Computer Center at that time?

Franta: Yes, with fondness too. I have allergies. So, one of the attractions was the computer center environment. I knew nothing about computers. No one did, so it wasn't surprising that I didn't. And the student who told me that they were hiring students to operate the computer, which means run the programs on behalf mainly faculty and graduate students in those days, also told me the place was air conditioned. And more over, they'll let you have a piece of an office, and it's air conditioned as well. It fit almost all my needs and now I'm going to get paid, wonderful because I then had to work my way through college, and I was going to have an air-conditioned place to sit as well. I signed up immediately. That was Marvin Stein who was doing the hiring at the time. He was the Director of the Computer Center, and it was in the building that is now torn down. It was the Experimental Engineering building, and it occupied the space where the new Computer Science building is today, name the name for me, please? Where Computer Science is today. What's the name of the hall?

Yost: It is—

Franta: Because I knew the man as well.

Yost: --Keller Hall, Kenneth Keller.

Franta: Yes. Keller. Thank you. Because I knew him pretty well as a faculty member. So that's where it was. It was air conditioned. It was a lovely computing machine. I was recently reading—now to back up a bit to make a certain parenthetical remark, as you don't exactly come into your little interview cold. You start thinking beforehand about what do you remember and is it correct and maybe I should just check on that fact or that fact, so obviously I did some of that between our set up weeks ago and today. I discovered lots of things. Among them, I'll call them 'The Rememberers' because that's what this is, a remembrance. And I discovered several things. Sometimes a remembrance that I now read is blatantly an error as far as my remembrance knows, and I found a few of those in things I passed through. And then I found several more that I suspect are in error but I'm less certain in my own mind than I was of the others. So, I meant to apologize to you if I remember incorrectly, I am to be forgiven because it is unintentional. But it happens. Do you realize that this is now over 60 years ago?!

Yost: Yes.

Franta: That's a long time.

Yost: I've done many oral histories, and that's part of oral history, is memory is not perfect, and historians use oral histories as an important source, but we also do archival research and try to triangulate between people we interview.

Franta: And by the way, thank you for the opportunity. I'm very happy to be able to do this.

Yost: Well, thank you. This is a wonderful opportunity for me and for the Computer Science and Engineering Department.

Franta: So now, carrying on—you asked about the computer, and I gave you the building. It filled a decent part of the second floor. The building was built so that you could run a huge crane down its length. So, it sat parallel to the street lengthwise. And there were steel girders, huge steel girders everywhere in this building. In the walls, between the bricks, and along the ceiling, and you could run a huge crane through this thing. I don't know what it had been used for before it became Computer Center building. But it was a rather homely building, and the second floor was occupied by the Computer Center. I guess according to a note I just read, it was called, when I started work there, the Numerical Analysis Center, but I remember it as the University Computer Center. And it was a CDC, Control Data Corporation 1604, which was a 48-bit machine, and I am now a junior in mathematics, and I can go in and play with this huge beast all by myself. I can manipulate every control. I can look into the back of the tape drives, where there were tubes in there that were 18 inches high spewing flames like one of the old science fiction movies from the '30s. So, it was all fascinating. That began then in the fall of '62 when I took this "computer operator" position. That began my education in Computer Science, which we may want to come back to because—with one possible exception, I never took a course in computer science. I took a lot of courses in mathematics, but we can come back to that too. So, I started programing as a student, a junior, in 1962. And two of the other undergraduate math majors also had positions at this computer center, and we had a little office enclave. And as I remember the staff was mainly students who ran the computer, and there was a Control Data maintenance guy who was responsible for the health and welfare of the machine. It used an enormous amount of air conditioning and it was really cold in "the computer room" even on the hottest day. And they employed, as I recall, two full-time programmers and they had influence on my development as well because one was pretty much interested in compilers, and the University Computer Center became known for some of these compilers. And the other was pretty much responsible for developing a vast library of statistical analysis tools. This collection of programs later became a system with a name and, I'm sorry, but I've forgotten the name, but it became a package that was distributed to a larger number of schools. The two full time programmers exposed me to many programming situations and had a very positive affect upon my knowledge base development as early as 1962.These influences remained relevant and significant even after my arrival in the Computer Science Department in 1970. There is so much that could be said about the machine and its programming. Every bit of it was interesting to me. There were basically three ways of getting it to do something. You could sit at the console and punch in octal numbers one at a time, or you could prepare a deck of cards as most people did in those days, that were entered into the machine via IBM card readers, or you could prepare your program as a roll of paper tape, and there were people who had rolls maybe six inches across, with each row of the tape being one six bit byte, yes, six bits per byte. And the only problem with the tape mode of entry was that in the winter time, before you got even half way through the running that roll through the tape reader, static electricity built up and brough the whole thing to a halt. You had to reload, restart the whole computer. Moving on, all was fascinating. Now, I started in '62, still in math, and from '62 on we had a fair number of visitors (to the Computer Center every year. For example, what could you study or learn from visitors? Well, you could sort of study the operating system but it wasn't always that interesting. You could study languages and compilers. Fairly interesting. You could study algorithms and how to realize efficient realizations of them, and so on and so on. For example, in those days, we were concerned with how long it took to compute a square root because if you might compute thousands or even several million of them in the course of a computation, and microseconds counted. Remember, in those days machine performance was measured in microseconds.

Yost: And computer time was expensive.

Franta: It was indeed. And I sure burned a lot of it and never paid for any of it. And so, I learned how to compute various functions. I learned how to go over and over the code for optimization. If you're doing the square root function, you didn't code it in FORTRAN or any other high-level language, you coded in machine language because it had to be quick, you played every trick you could think of to optimize its performance. One of the two other programmers I mentioned, a fellow named Larry Liddiard, who died only last October (2019), spent two weeks, I think, working on square root. Every time you could shave off a microsecond it counted. We worried about things like that. I started reading the literature. I mean, in those days, in '62 right up until, to '70 when I joined the Computer Science faculty, you could, I could, study mathematics full-time and still be full-time up to speed on computing. There wasn't that many people or journals to track, right?

Yost: Right.

Franta: So, I read a few authors in a few journals, or something like that, and I had it all. And, the journals were available in the Computer Center, to read for free. What did we learn? Well we had FORTRAN, obviously, probably the best one in the country because Larry developed a very nice compiler. We had SNOBOL. We had a visitor from ETH in Switzerland who as a guest of the Computer Center, or just possibly the Math Department, offered a formal or informal lecture series on Algol, which is a completely different animal from FORTRAN and as a result vastly enlarged my knowledge base. And so, there was already things going around in your head about, what does this mean, and what does this different organizational structure mean? Also, we weren't deprived of languages. We had and could run experiments using FORTRAN, Algol, SNOBOL, COBOL, and then on to using statistical and other packages and writing code to numerically solve PDEs to experiment with operating system behavior. And then Assembly language. And so, it was a wonderful time. Everything was new and fascinating. I was learning as much about computing as I was learning about mathematics. And I realized by then, that I was fairly good at mathematics, and so it was the path of least resistance to study it. I was getting straight A's as well, especially in graduate school. So, it was the right move to make. I am a well-planned individual. Every move I've made has been well-planned and then executed. Ha! Hardly! I was lucky to be able respond to opportunities as they presented themselves.

Yost: And you went on to pursue a graduate degree, a Master's in mathematics?

Franta: Yes, I worked part-time, sort of on an hourly basis, I don't know how we kept score, until '64, June of '64, when I took my bachelor's degree in mathematics. And Marvin Stein came to me, I think it was Marvin, it could've been Bill Munro, and said "You know, we would like you to go for a Master's degree in mathematics, and you can work here and just go to class when you have to and do what you have to," and so I did. So then in June 1966, I got a Master's in mathematics, I think it's a Master's in Science but it was a major in mathematics, and then one of those two gentlemen came to me the second time and said, "You know, why don't you just go on for a PhD in math and we'll hire you full time." So, I was a full-time employee at the University of Minnesota from—one way or another—from 1966 until 1982 when I left the Computer Science department. So now I'm getting paid. I'm getting benefits, I'm getting money put into a retirement program and I'm going to school. I mean, I'm in heaven. And during this time, because we are now in 1967, the year after I received My master's, we got the Control Data 6600. Now, one of the departmental documents from their 40th anniversary says that the Cray was widely considered the world's first supercomputer. I very much find that an error.

Yost: I disagree with that as well—CDC 6600 was the first. [While no historians now see the 1604 as the first supercomputer, there is vigorous debate on whether the 6600 or IBM Stretch a few years prior has the distinction. I believe the more powerful 6600 is the first.]

Franta: There were people who during those early years thought the 1604 was a supercomputer because it was for scientific calculation, numerical calculation of any kind. It was better than anything IBM built. And Burroughs and PDP-11s didn't really count in that sphere. So, while some thought the 1604 could be considered a supercomputer—and they did when I started in '62, in my mind, the first true supercomputer was the CDC 6600. But this time most not just some considered the 6600 a supercomputer. And so now (1967 and beyond), I'm doing very well in the mathematics, I am learning a great deal about computing, and I'm enjoying myself a great deal. In mathematics, some of what we studied was so beautiful I was amazed, this being especially true of functions of complex variable with results like the residue theorem. As an aside, today I'm back to doing mathematics and computing because I'm home, right? I'm retired. I can do what I please. I'll tell you at some point if you're interested, in what I'm currently pursuing in mathematics and computing. And so, getting back to 1967, people are wandering into the Computer Center from all over the world because we've got, as my memory tells me, serial three, the third constructed instance of the 6600. But for some reason I'm now, not sure my memory is correct, as one piece of data I found doesn't jive with that remembrance. I think CERN had number two and Lawrence Livermore had number one. But maybe I have that a little distorted. We had an early version, and because it was so close, all the Control Data people developing compilers, operating systems, da, da, da, used our machine at night. Well when did I get to use this machine? Every night. And so, I had all this inter-mingling with the compiler people, the operating system people, the visitors from everywhere including ETH and so on. The Computer Center and computer room was a beehive of activity and a place for discussions and learning for me. Everywhere I went I was learning something new. It was lovely. The theory of complex variables during the daytime was fascinating me because I loved it, it was pretty stuff. At night I was learning more and more about algorithms, compilers, operating systems, etc. So, I was having a ball. The 6600, you probably know, was not installed on campus. The 1604 was located in Experimental Engineering where Keller Hall now resides. The 6600 was put off campus. Those were the years of campus violence and destruction of computer centers and the Vietnam war, et cetera. And so, it was located in Lauderdale, which is a suburb of St. Paul, located northeast of the campus on Highway 280, which may not now mean anything to anyone. It was located in an unused Northern States Power Company building and, as a consequence, it had an enormous amount of power fed to it, and it had access to an enormous amount of access to water because cooling a 6600 was a task. And it, the 6600, filled a room twice the size of the 1604. And so, we were out there every night for a couple of years doing various things trying to make the machine behave. Control Data very much got the hardware right. They had a lesser clue to what they were doing on the operating system side. That's an overstatement, but you get the idea. The hardware architecture and performance was grade A. The software architecture was good, but the performance of the operating system was a D+. And so we spent a lot of time with the Control Data guys, one guy in particular named Greg Mansfield, and I have no idea where he might be today, thinking about and discussing the operating system—and he ultimately nearly single handedly redid the operating system and made a thing of beauty out of it. We got to be pretty good friends. I spent one year in—Christmas in London- with him and I don't remember why, but we were both there for something. So, I was having a ball learning an enormous amount of stuff. In various ways I was contributing to the 6600 platform. I was quite happy. To continue my story, I took all my coursework in mathematics, never any in computer science. I took courses from Bill Munro in numerical methods in mathematics. Bill Munro was a sweet and very smart fellow, and Marvin Stein's partner in authoring one of the first books on computer programming. I also did programs for many numerical solution problems on my own. It was fun and I had access to the 6600. Recall I started solving partial differentiation equations on the computer in ’67, just as a fun thing. And during this time, I decided that, I might actually get a PhD in mathematics because I had taken up enough courses, so I took my written exams in mathematics and then my oral exams in mathematics. You had to be able to read mathematics in its originally written language so I learned French and became quite good at reading mathematics papers in French. Never pronounced some of it correctly but could read it. And as I said, I took my oral exam, your pre-lim oral exams, in mathematics. But then my interests kind of shifted. I was interested in Monte Carlo stuff. I’d always liked statistics too, and we had a guy named Bill Lindgrem, in the math department who was a first-rate statistician and a really good teacher. He wrote a great textbook on statistical methods and did a lovely job of teaching a few Monte Carlo things. I started reading von Neumann and doing some Monte Carlo things on the computer, and to make this story shorter, I ended up with a dissertation, a PhD dissertation, that was Monte Carlo-ish that modeled Stochastic systems, but it was—I was applying Monte Carlo methods to computer systems behavior models because they were pretty horrible performers in that day for lots of reasons. Not just Control Data machines, but other vendor machines as well. They just didn’t perform like people thought they ought to because the software was not in harmony with the hardware. And so, I started doing modeling of computer systems, and doing systems and their Markov-related things. And I didn’t—it didn’t seem reasonable to present that dissertation to the math department because they didn’t care for it, at least that was my belief. It was presented to the—in ’68 or ’69, I guess, to the graduate faculty of Computer Science. The computer science graduate program which then existed, as this little notebook says, the first PhD went to a guy named Dwayne Zimmerman. I knew the name, but I never met the man, so I don’t know what became of him either. So, I think I was the number three, again, I could be in error, PhD graduate from the graduate program which was—which existed in ’69. So—

Yost: And that program started in ’67, is that correct?

Franta: That I don’t know with certainty. I think that’s right because Dwayne Zimmerman took his degree at least a year earlier than I did. Officially from a graduation point of view, it was August 1970 for me, started working in September then on the faculty, so he likely got his in ’68 or ‘69, so ’67 should be correct for the beginning of the graduate program, I do not remember very well, and then joined the faculty. I had not expected to join the CS faculty. I expected that I would go elsewhere, for example, Control Data or whatever, yet I was presented an offer to join the CS faculty, I guess computer literate candidates were hard to find in those days. Now not having taken any computer science courses, I was still as up to speed as anybody on the planet at the time on what you could and couldn’t do and what the interesting problems were. Now, I didn’t choose to be—I didn’t want to be a CS theoretician although some of my research with my graduate students in later years migrated more to a more theoretical realm, and a lot of math. But in the early days I was interested in this Markov, Markovian behaviors in modeling and simulation and languages and operating systems. And through the years, I tried to make a list. I did research on imaging, operating systems, languages, computer architectures, distributed computing, Monte Carlo methods, digital simulation, and then I want to, and will come come back to, computer networking where I spent a great deal of time. So that’s how I spent my research time while in the Computer Science Department.

Yost: Just so we have it on the recording here, your dissertation was titled, “A flow-oriented computer system simulation language with applications”.

Franta: Correct. Yes. I think that’s right. I still have copies. And so, that’s a modeling and modeling tool dissertation.

Yost: Can we take a step back? I'm curious as to what level of help and service the Computer Center provided to faculty and to what extent they had to learn to program and to what extent they were kind of instructed in how they might apply computers to their field of science and engineering or perhaps the social sciences as well.

Franta: I'm not going to be able to provide a reliable answer there, but I can tell you this, going back to my days on the 1604, some of the major users, because I ran all the programs, you know, I was part of this little staff that fed the computer card decks, many of those were from the social sciences, economics, and I remember, in particular, that one of the routines on the 1604 had gone bad, and was producing inaccurate numbers. And I tried to go and find all the people who had run programs that were disadvantaged by that, and many of them were in the social sciences. There were a couple of medical accounts, from the medical school. There weren't—I don't remember any from the business school in those days, but they had a little machine of their own. There were three machines, at least, on campus. One was in the medical school. I think it was a 3800, a CDC 3800. And there was one on the west bank in the business school, in the Carlson School, maybe it wasn't Carlson in those days, that I think was a CDC 3600 or something like that. So, there were maybe three computers on campus—they were all of the CDC 1604 Canon. Same architecture, same hardware plan but only one 1604. So, I don't know how most of the people learned. In the early days I am not aware of any organized program to teach people how to program. Once the Stein and Munro book was available Bill Munro taught a course based on their book, as a math course if I remember correctly. Once the undergraduate program in CS started there were many courses on programming, and sometimes nonstudent people "sat in". If you want to know pre our undergraduate program, the graduate program faculty was in the main Stein, Munro, Leavitt and Frankowski. Stein and Munro had written a programming textbook and taught programming, as did Leavitt and Frankowski.

Yost: Yes, they did, early and very influential texts.

Franta: Jay's thesis was numerically oriented. He was solving PDEs so I'm assuming he knew how to program in FORTRAN. Same goes for Krzysz, although I do not clearly remember. Recapping a bit, Marvin Stein and Bill Munro wrote a book together called *Computer Programming: A Mixed Language Approach*. Nicely done too, I know, I read it. But I am not aware of either of them ever writing a program. I think their book was the second serious treatment of computer arithmetic and programming written. There was a man named Ivan Flores, a private consultant with an affiliation with some university [NYU early on in 1963 and 1964 and later Stevens Inst. of Technology, and NJIT], that I think he wrote the first textbook on programming. This was on software and coding, high-level language, sort of like what Stein and Munro did but a couple years earlier. I think his prior to '64, and I think the Stein and Munro book was '64, and so based on 1604 concepts, but oddly enough, my copy has been lost. The faculty in fall of 1970, at introduction of the undergraduate program, could all program.

Yost: Do you have their names?

Franta: Yes! In total there were, by my remembrance, eleven of us. By name, the group included Marvin Stein (head), Bill Munro, Jay Leavitt, Kris [Krzysz] Frankowski, myself, Don Boyd, Al Hanson, Oscar Iberra, Kurt Maly, Peter Nicholson, and Sartaj Sahni. I apologize if my memory has omitted anyone. All could program and the list expanded our language and discipline capabilities. Leavitt, Frankowski and Munro covered numerical methods and basic programming. I covered modelling, digital simulation and Monte Carlo methods, and sometimes other things. Iberra covered automata theory and other theoretical considerations. Sahni covered data structures and associated algorithms and complexity, Boyd covered programming and operating systems, Maly covered data structures, programming and programming languages structures. Hanson covered artificial intelligence and associated programming languages and techniques. Finally, Nicholson covered, many things, including operations research, programming and an item I hope to broach in a bit. Some of these people only stayed a year or two, others stayed a long time. With this crew we had, in 1970, enough faculty to offer a goodly number of courses on a goodly number of topics. Additionally, faculty had time to do considerable research, write papers and publish books. I, for example, wrote and published my first book, entitled *A Process View of Simulation* in the middle of our hectic initial growth period. So, this was the crew. This document that I pointed to, again says the department was known for numerical analysis. I'm not—we had some very apt, capable mathematicians that could do that, but I don't know that we ever did much of that. You know, Jay and Krzysz weren't big into publishing, I don't know why, I never asked them, but they didn't tend to publish much. Bill Munro and Marvin Stein weren't publishing anything because they were up to their eyeballs in getting this thing running. And so, as a department we published on a variety of other subjects. We had a smattering of other things. I was allowed to teach courses on Monte Carlo methods, in fact, wrote a book, my first book, on the subject, as mentioned earlier. Kurt Maly was into languages. His thesis advisor had developed a thing called SETL, meaning Set Language, I think. It was a language based on set theoretic concepts and we actually did some analysis in terms of language construct comparisons about what could you do better on one form of language what on another, including SETL. Most of all, I'll tell you something specific, we had, in Monte Carlo simulation there was the thing, data structure, known as the event set. You have to keep track of when events are scheduled to happen and you have to be able to insert them into the right order and remove them from the order and change the order, dah, dah, dah, and so what's the data structure that best serves that need? Well, the conventional one was what I called a double-linked list. That means nothing. Doesn't matter. So, we explored, that is as one research project we did, Kurt Maly and I did, is what's the effect of using other structures, like some of the SETL constructs, that set theory construct. We published a couple papers on that. So, research of a language, data structure, operations research sort, hardly numerical analysis. Iberra was into automata theory and nothing else so his publications were on automata theory, Turing machines, etc. Hanson published on AI subjects, Sahni on algorithms, complexity subjects, and a nice book on data structures. And so it went, lots of stuff on lots of subjects, certainly not all or even much numerical methods. Then when the years went by, I got off into all other kinds of things. And so collectively we had artificial intelligence, we had languages, we had some operations research, we had numerical mathematics, automata theory from the formal sides, so you could learn about and conduct research on a number of topics. We covered all of this, between the eleven of us initially, then more as the faculty grew, the coverage expanded more. But, even initially we covered a reasonably rich set of subjects from the first year, and it was a hell of a lot of work because, because, you know, what the hell do you use for a textbook, you know, what's out there? Or should I write one?

Yost: Right.

Franta: And so, in terms of the modeling or digital simulation, I finally just wrote one. And so much for the first year or two. Now if you're interested in somewhat of an aside, Peter Nicholson again, in that first year—by the way, when I did the short video for the Computer Science Department, they asked me if I thought I was doing anything that was truly significant in the early years when I was working in the Computer Center, and here's what I referred them to, and I thought of it yesterday and meant to just mention it to you. There's a book, this is a book called *An Introduction to Mathematics* that's written by Alfred North Whitehead, who was Bertrand Russell's thesis advisor. This book was written in 1911. Well, I was so in love with mathematics that I actually read this as an undergraduate. But there's a line in there that says, "Civilization advances by extending the number of important operations which we can perform without thinking about them." So, in my little mind, computing did that in so many spheres, that this was my excuse for thinking it all worthwhile in those early years. But I just wanted to mention this. Now, going on to the Peter Nicholson story. During the first years of having an undergraduate program we had a major onrush of interest in our program and computers. We had a lot of people who asked about programming skills, it's kind of in that vein but different. In the first year, 1970, we had an in-rush, we had a lot of students, so even in my modeling class, I bet I had 60, 70 students for every offering, every quarter. First level programming course could have a hundred easily. We were also offering these things on television in those days. So, we had students out in IBM Rochester, Honeywell, 3M, and we were doing these on television, and it was interactive, and it was kind of nice for me because, I'm doing all different sides of things. I'm sorry if I—

Yost: No, no. This is fascinating, please continue.

Franta: The way you worked on the TV is sort of like we are right now on Zoom. I—there was a camera up here and there were various cameras in the room, and after my first two offerings of the modeling class, I had a pad of paper, it was on 6 by 6 inch pages and 5 inches high. These were little blue pages that went well in the camera. And I wrote all at—the first, every time I did it the first time, I wrote the notes that I put on a blackboard on these pieces of paper. So, by the third offering, I don't have to stand and get full of chalk anymore and make a mess, I can sit at this desk and use these pieces of paper, and I don't have to write them anymore. Now I get to talk about them instead. And then I get, "Slow down! Slow down!" They're, the students, were trying to write them down as well. So, I liked the TV, and we even did some at night. So, I had a room full of students and then various numbers on TV. I know we had IBM Rochester, and we had 3M and we had at least two places at Honeywell. We may have had more, I don't remember. But that was an important, you know, learning mode where you didn't have to come to campus. That was circa 1970!

Yost: Right. Do you happen to know if Marvin Stein and others who led the way in forming the department visited other early departments and learned from them and were there any particular schools that Minnesota in part modeled itself after in CS?

Franta: I don't—let's see. The very first computer science department was…

Yost: It's disputed. Stanford and Purdue both lay claim—

Franta: Yes, I think it was Purdue. And I think Peter Denning was the department head, he happens to be a friend of mine and I haven't seen him in years but—

Yost: I've done an oral history with Peter Denning. He's been a great friend to CBI, and Dorothy Denning also, who I interviewed.

Franta: Yes. I noticed that in your list. You missed Grace Hopper though! We can come back to her. I have stories about her too. There was a great deal of interaction between Purdue, NYU, us, Illinois, and several other schools from the 1604 days onward. So, it would've been almost impossible not to have discussed the subject with others, you know, even I was in contact with people at these places. Marvin and Bill surely had more contacts than I did. My first excursion abroad was in 1967. I was a graduate student as you now know, having just received a master's degree, and I was now sent to Switzerland because there was going to be the first conference on supercomputing. And I was sent to gather information on what people were doing. Why I was sent, I do not to this day know, but Marvin picked me, and then, as long as I was going, I had to go to England and go to [RAF] Harwell, which is their atomic energy center because they were in the age of supercomputing and what could I learn from them? And so, I did some personally and so I have to believe discussions with others was on a fairly large scale by Marvin and Bill. By the way, to go back to the 6600, and the statements made, this is from the Wikipedia article. Everybody knows that Seymour Cray is the architect of the 6600. Now I knew Seymour Cray fairly well because I was at Chippewa Falls for many reasons during the ’67 to ’70 years, in particular, dealing with things for our machine. And even this article points out, his cohort in crime for the 6600 was a fellow named Jim Thornton. Now I don’t think you ever interviewed Jim Thornton. I missed it if you did.

Yost: We do have one with him.

Franta: Jim—here’s my belief about how it worked. I believe Jim Thornton was the architect, and I could’ve asked either one of these guys many times and never did it. My belief is that Jim Thornton would be the architect, meaning he sets the rules about how the processor was going to work, how the memory’s going to work, you know, it was fairly innovative ideas in the thing about the eleven machines running in parallel, really there were only two, but it looked like eleven machines, and how you could divide memory up and have more than one job in memory, so that while one “job” was doing I/O [Input/Output], you could do computation on another one, it’s already in memory, just have to shift a few parameters and you’re into the next one and back and forth you could go and they had seven called control points. That’s all Jim Thornton, and others, I think. And what Seymour was, was an absolute genius. If you’re going to make this machine this fast, and you’re in discrete transistors, there’s a lot of heat. So, to make it fast, you need a lot of components and you need very short wires. So, Seymour’s major contribution to this thing was how do I keep this thing from melting onto the computer room floor, and it's absolutely brilliant how he did it. So, he's a packaging genius, in my view, and Jim Thornton was an architecture genius. Should I be wrong I apologize to all. I'd like to come back to Jim Thornton later because he played a later role in my life related to my activities in computer networking. Anyway, to go back to the first year, we had all these interactions, and we had a group of people from everywhere, who asked about how we learned to program, we had a bunch of people also come to us, me in particular, and I don't know why I was singled out, saying, "You know, we're interested in understanding—we don't want to program—but we're interested in understanding basically what this beast will do." the beast, in 1970 would've been the 6600. But we wanted to know what this can do and what the limitations are and in some way that we can understand what affect it is going to have on my life. And these were social science types, medical types, some business school types. No mathematician. And so, in 1972 we even—it turns out, what I'm about to tell you, is even recorded in a paper written on it. I didn't know this paper even existed—I just found this yesterday. Apparently, we wrote it. The things I forget! We wrote it in 1972—it was presented in 1973 at an ACM conference, and it's entitled, "The impact of a *Computers in Society* course from a student perspective". So, that's the way it was written. But what we did was we put together a course for the world at-large that was called "Computers and Society". And what did we do? Peter, this was done by me and Peter Nicholson, the fellow mentioned earlier. What Peter and I did was to give a few soft-core lectures on how this thing worked, you know, in very general terms but we were fairly effective in conveying what it could do, how fast it was. And then we would invite a score of outside folks to give lectures. We even had Grace Hopper there once, and she used to carry around these eleven-inch pieces of wire. That's how fast, that's how far a signal could go in a nanosecond. And she'd use that—you don't have a video of Grace Hopper, and that's too bad. She was quite the character. Admiral in the Navy. Did the first COBOL compiler. And so we put together this course and the way we did it, we gave the first lectures on how the computer worked, what it did, how big it was, why it had to be cooled so much, da, da, da, da. And then we invited in a whole bunch of external speakers, and, for example, we had Grace Hopper up to talk about her view of where computers were going and what eleven inches of wire meant, and why it was important. She was a hoot. And then we had futurists. Some of them resided in town. There was one at Sperry at the time, I can't remember his name at all, but I'll think of it, and so on and so on. So, I'd say we gave, Peter and I gave less that 25 percent of the lectures, and we organized and brought in all these speakers for the remainder. The course went mad, I mean, I think we had 50 or 60 people for the first offering in '72. I think it says the spring of '72. And the year later, we didn't want to do it anymore because we had hundreds of people who wanted to get into this thing. And I don't actually know how it was actually killed because Peter and I went onto do other things and it continued in some fashion, and then it was gone before I left the university. So, I don't actually know how it was killed. It attracted a huge number of people and it—the futurists were giving predictions about because the students were interested in, "Well, how's it going to affect my life in the future? What does this mean?"

Yost: Right, can understand the enthusiasm for it.

Franta: And my remembrance is that they, all the lectures, they were in retrospect ineffective. My remembrance is that nothing that any of them said ever came true, and if it did, it was insignificant. So, I began wondering why was that the case. And then it sort of dawns on me that while maybe not a strict instance of, it's a Thomas Kuhn discontinuity in the way things were done. And so, all the lectures were going along the technology curve that existed. These were advancements of the model that existed instead of exploration of the model that just exploded in front of them and how it existed, and nobody knew how it was going. So, the course was interesting. People loved it, but none of it turned out to be true. And I believe now in retrospect it was because of the Thomas Kuhn affect. I don't know if that is of interest or not but—

Yost: Yes.

Franta: We did that and there's a record of it.

Yost: As a historian of science, of course, I've read quite a lot of Thomas Kuhn and his work on the history and philosophy of science.

Franta: Yes, I would assume that you had. This is, if you're interested in, yourself, this is in some, I can give you the reference. It's an ACM symposium from '73.

Yost: Great. Speaking of the ACM, when did you join the ACM, and can you talk about how professional organizations, the ACM and maybe the Computer Society as well, influenced you?

Franta: What do you mean by the 'computer society'?

Yost: The IEEE Computer Society.

Franta: Oh! Ok.

Yost: Don't know if you belong to that? Some people belong to both. Some one or the other, in academe both prevalent, but I think ACM a bit more.

Franta: Well, in my day, I belonged to both. I received, you know, I published, let's see. Let's back up with this thing. As I suggested to you already, there were, those societies and their publications were important to me because, even as an undergraduate, I was learning from them, right? They were laying around the computer center so that's where I started reading them, and as I said, you could track the world at large because there wasn't that big a world in those days. There are authors that you want to track and the rest you could forget. They became very important once on the faculty for a number of reasons. That was the outlet for your work. And, you know, if you published in an IEEE or in ACM journal, that carried more weight than publishing in XYZ journal of fun facts, and so those were frequent outlets for the papers you wrote, you know, I have an a number of papers in both, in various journals, of both the IEEE and various journals of the ACM. And then some other journals some, you know, *Acta Mathematica*, which is out of Europe, out of Germany, I believe, is one. So, they played an enormous role in, as an outlet for the recognition you needed to advance as a faculty member, your research work. They were equally important to keep up, provide ideas, I used to, have a tendency to read those articles that relate to the field you're doing work in. But I decided at some point it was also equally important to at least scan the things from fields you weren't doing anything in because you might get an idea. You get something that you hadn't thought about in that particular way. Remember, my whole life, the theorem has been it's all in asking the right question and that's so true in mathematics too. It's how you formulate the question you're interested in, you want to—how you—if you have a problem to solve, it's how you formulate the question. And so that's one of the theorems that I've carried through life. So we've diverged from that point. To return, I belonged to both societies. I remain a life member of the IEEE. And I don't have any significant reason for having kept that one and letting the ACM one drop. I used to love ACM *Computer Surveys* because they were tutorial kind of articles, so I had a large collection of those. Published a lot of articles in the journals and the *Communications of the ACM*, think it still exists, I haven't seen one in a while. But for some reason, I just kept the IEEE one and now I'm a life member. Once you reach my age, and you're still above ground, they provide a life membership. So, I get free access to the journals, and, you know, anything else that they offer, health insurance, da, da, da, da, at reduced rates. Travel. They do tours that they put together. There's no particular reason for it. I found them both important. Just happened to retain that one instead of the others. I still read the *Computer Communications* magazine. You can get it online now and so you can just download it as a PDF file. And I have access to all the history. And so, you know, that's enough, I guess.

Yost: So, you've talked a little bit about how the courses grew and the number of undergraduates escalated for many of these early courses. Can you talk about the origins of the graduate program and setting up a graduate study, funding labs, the importance of sponsored research, grants to that and what you specifically sought with regard to sponsored research?

Franta: Well, again, it's one of the measures of success, right, as a faculty member. There are several routes. There's the, my mind's gone blank, the normal one is the one out of Washington—

Yost: The NSF, National Science Foundation?

Franta: The NSF. Yes, thank you. I don't know why my mind collapses, like a quantum state and then it collapses. I didn't—I took less advantage of that route. I had, let's see, how do I put this without sounding outlandish. First of all, from 1970 until almost until the time I left the university, I was a regular consultant at Honeywell Systems and Research Center. I was a consultant—well, I was a consultant to about eight or ten companies. So, the ones in town were Honeywell Systems and Research, that's the one on Stinson. I don't know what they do there now. It was a beautiful building. It still had signs on the wall that said "Loose lips sink ships" because they made bomb sights there during the war. I consulted for 3M. I consulted Network Systems. I want to come back to a little later, that was with Jim Thornton, and Ford Aerospace and several other places. And so, I was rich in money from industries. I made less use of money from NSF or other government agencies than did most of my colleagues. I also was an associate director of the Computer Center while on the faculty. I had a lab in Shepard Hall. And we did, among other things, imaging work, and I have a story there I want to tell you about, one of the things we did. And so, I had access to students that I could hire on university money as part of this lab too. So, between the industry people and the computer center relationship that went on, I had sufficient money. I funded—can I leave the desk? Can I leave you for five seconds? You see that? (Visual-Indy car photo).

Yost: Yes.

Franta: I funded that and my racing career via consulting. I had access to a lot of funds and a lot of people. And then along the way, you know, I spent a semester as a guest to the Heinrich Hertz Institute at the University of Berlin. So, I lived in Berlin for a few months as an adjunct or whatever you want to call me, from the University, when the wall was up, the Berlin Wall. Lots of stories from there. And then I spent a period living in Haifa, Israel as a guest of the Technion. So sometimes I was gone and still had these things funded like, one of the things I wanted to talk about later was funded by Network Systems. You know, I had a, I funded graduate students that way through some of these relationships. Although funding is important, otherwise you can't support graduate students very well. More than twice I had people who didn't care about funding, they were usually from Israel, had been in the military and been compensated in other ways by them. But otherwise, very important. Does that sort of answer your question?

Yost: Yes, it does. Can you talk about, a little bit about your educational philosophy and management style, working with graduate students?

Franta: Well, one of my graduate students was Elaine Frankowski. She was one of my PhD students. Now, why are we singling her out? Because she used to run around telling people that I knew—she'd come in with various routes she thought about taking. And I'd sit there and listen and say, "Don't do that. Try this one." And so, she went around telling I knew what's not going to work before it was tried. So that's a management style. I was very lucky. I had a collection, I think from 1972 to 1981, of around 15 PhD students. I have no idea how many master's students, and they were all pretty mature individuals, so we got along very well. I'm still in contact with a couple of them. In fact, one of the trips that I was going to take this coming fall was to visit one who is now living in Bratislava, Slovakia, and is planning on moving to Israel, but this COVID business has brought that to a halt. There's a couple that I still actually know. They were all very mature, and I attracted somehow, some pretty bright individuals, and one or two at a time was about all I could handle with all the things I was engaged in. So, the management style was let them flounder for a little while on everything, I guess. And then maybe make a suggestion like in direction or approach, to get them pointed in a direction that I thought would be fruitful. Sometimes I help them, but I tried to avoid, I mean I actually did part of it sometimes, but I tried to avoid that action. And then I made them write it up so we could submit it, and sometimes I rewrote it. But sometimes I didn't. I don't know if that's what you mean by management style.

Yost: Yes.

Franta: I was very lucky. I had a very good group of people. Some of them went off to be professors in various colleges or universities. Some of them went off to be big time in universities, and not—let's see, I don't—some of them went to industry but not too many of them. This was always a struggle, one of the reasons I preferred the non-theoretical side of things. If you're going to focus on that, my view was that you are going to end up at a university as a professor, and not all these people wanted that so I had to do something with them where they had a hope of getting really good employment because maybe they wanted to do research maybe in a government lab or in a corporate lab or whatever that they chose but I had to have something that was usable in the world. So, I was careful about that. Now, that doesn't mean we didn't do a lot of—we published a lot of papers on various networking protocols, access protocols for wireless networks and the like, some of which still have never been tried, but they are fully mathematically analyzed because that was both fun, interesting and may turn out to be useful to somebody at some point in time. They are all in journals so, you know, it's a combination of utility and what you think you must do with writing at the time. Does that help?

Yost: Yes, great. In the mid-1970s, you're publishing in the simulation space, and in 1977 you published the book *The Process View of Simulation*. Can you talk about that area of research in that book?

Franta: Well, in some sense, I got into that because I was looking for still more general platform for doing any kind of stochastic modeling. And that book explains how you can take that, this Simula system, as a base and apply it to queuing theory, operating systems, apply it to any kind of system modeling you want. You could, we could model a molecule. It was well structured and available on a large number of computing platforms—but then I also did it so I could realize all the theoretically fashionable constructs for operating systems. They're also a part of that book as a guide to how might I use such constructs, to model any kind of system, and then simulate the operation of that system, and I could certainly use it in modeling an operating system which was of considerable interest in those days. The approach got so successful because the Simula system was a proper base for what I wanted to express. Simula was originated by two fellows in Norway, Ole-Johan Dahl and Kristen Nygaard. They wrote a small amount of machine code and then they wrote the rest of the Simula system in the Simula language itself. They only had to create a small, little piece of code that would then sort of compile or be interpreted on a given machine, the rest of anything else you wrote in that language. So it was written, more or less, in itself. And what I had done with it was considered neat enough that Dahl came to the University of Minnesota to meet me. Mats says that those two guys, they are both deceased now, Kristen Nygaard and Ole-Johan Dahl, and I guess they are both deceased, but Mats says that they both got some posthumously delivered award for having discovered or developed Simula. Well, anyway, one of the two came to see me, and I became editor of, in Minnesota, of the Simula journal or newsletter for a short period of time, I think year and a half or something. So, it was a neat way of describing systems in order to simulate their behavior. It ran on a 6600. Ran very well. If it was among the compilers you could get, this one worked, partly because it was mostly written in high-level language itself. These guys were pretty sharp. And it had all the statistical tools you needed in there through the first 'tranche' of what you might need so you could statistically monitor convergence and how long you had to run it, what to look for and so on. So as a book, it did very well. Did better than I thought it would, could. And it was fun to write. I had a good time writing it.

Yost: And in 1981, you published *Local Networks: Motivation, Technology and Performance*, and that was one of the first major books on local area networks. Can you tell me about that research and book?

Franta: I think it was, I think it was the first.

Yost: Ok.

Franta: Yes, if you're going to let me go there, that's a long story, and that brings all kinds of things together. Before I do that, if you'll let me have one interlude?

Yost: Sure.

Franta: Ok because the interlude is the follow-up. The CDC 6600 was a major success by any measure. I think it sold over a hundred copies at some exorbitant price, who really knows?

Yost: And I think CDC figured that they would have thought they'd be OK [on their R & D investment and pricing] if they sold 10 or 15 of them.

Franta: Yes. So, they were pretty much in every university that wanted one. They were in every government lab, they were in most of the labs around the planet, CERN, the communications centers, the laboratories, nuclear and otherwise, energy labs. But Bill Norris had this vison. You know about the Plato system? Have you ever—

Yost: Definitely yes. We have the Plato records [part of the Control Data Corporate Records collection as well as the William C. Norris Papers] at CBI.

Franta: Do you? Well, let me add to your stories. So, in terms of the video, should I say some word about what—yeah, let me say a word about what Plato was for. Plato was a spin-off of an educational system that was originally developed at the University of Illinois. And they ran it on a machine that they built there called ILLIAC and I think there still are six or seven versions of it still in use. And what it was, it was computer-based instruction. And so, Bill Norris, because we talked about it, took this on, on the 6600. So, the Plato system version, something like version four, I believe, was transferred to the 6600. This was his second major market that he wanted to conquer because, you know, there is a cap on that science and computing market, you can replace many of them with the same animal, so there was a

7600, da, da, da, da. But he wanted another market, and this computer-based instruction called Plato was his second market choice. And it was pretty interesting for its time. I mean, it was terminal-based. You could converse with your colleagues through the mainframe and so on. Well, about 1979—and this too is documented, there's a paper on it. The paper's called "Joyce: A next generation personal computer". And this story hasn't been told, I don't think any of your other invitees have told it.

Yost: I have not heard of it. I found that intriguing when I saw that.

Franta: This was published in 1980 it says here. I thought there was more than my name on it so I don't know why the copy I found on the internet has only my name on it but, because I thought the fellow who did most of the work, we did it together, is on here as well. A guy named Rudy Weinburg who was one of my PhD students. But anyway, we built this machine, it would be called a workstation today, it was called a next generation personal computer in 1979 and when it was published in 1980. And we took, we found components for the bus, the architecture, the disk, the memory, the CPU, the biggest, fastest things we could find, and we built this desktop computer. And what did we do on it? Well, through someone else's interest, we took the Sanskrit course off the Plato system, and I can't remember if we put all of it or just a significant piece of it on this platform that we built. Then we showed it to Control Data. They were not well-pleased because this little guy could do everything that the 6600 could do. And think about the consequences here. Now you could take 6600 and make it a warehouse and then download the lesson plans to these individual, cheap machines, and you could scale this up much beyond what you could do with plasma terminals like the 6600. They did not like it at all and that's the end of that. Add that to your collection of stories.

Yost: And was Bill Norris at the demo you did?

Franta: I tried to think about that yesterday. [Robert] Price was [Norris's successor as CEO and part of the 3-person leadership team under Norris], I'm sure, but I don't remember with any clarity anymore. This was near the ending of my tenure at the University, I was about to take leave. I remember—boy, I shouldn't tell, I was asked to leave behind all the documentation for this so that I wouldn't go off with it. So that's a story you don't have about Plato. It was a very effective system for displaying interactive lessons on and, you know, Sumerian wedge and crescent is not easy to—not of interest to most people but it was interesting to somebody, that is the one we picked it for the demonstration. Oh, by the way, just so you know it, I came equipped with my core memory. So that's the end. Now, redirect me again—oh you wanted me to talk about the networking stuff?

Yost: Yes, you had, you wanted to talk about Thornton—

Franta: And my book. Somewhere during my heyday, I used to get venture capitalists coming to me, asking about technologies. This was a pretty inventive period; the Twin Cities was a hot place. So not only were we the hot bed, you know, I was very fortunate, I'll say this up front, I got to be a part of supercomputing when it was burning, sizzling hot, and I got to be a part of networking when it was burning, sizzling hot. So, I got to play two rounds. So somewhere during my tenure on the faculty, a venture capitalist came to me and he had this business plan for a networking company. And the venture capitalist was Norwest Growth Fund, which was, Norwest went to Wells Fargo, I believe.

Yost: Yes.

Franta: Or is it US Bank? Wells Fargo?

Yost: I think it was Wells Fargo.

Franta: It no longer exists. They were the biggest venture company in the Twin Cities at that time, and they asked if I would look at this thing. Well, I looked at it. What was it? It was what we would then have called a local network, although it wasn't called that in the material, and it was being spearheaded by a fellow named Jim Thornton. Now, Jim Thornton's the same guy that I represented earlier as probably the chief architect or the architect of the 6600 and he did the 6400 and some other things too. And he was trying to put together this company for this kind of network. And so I said it looked really interesting, but I was not a venture capitalist, although I was later, but I said I thought it had merit and that they should give it very serious consideration. And time went by, and so, now we're a couple years later. They have been funded. They have developed products. These are all ex-Control Data people, so they were supercomputer engineers, da da, da, da, and Jim Thornton, who was the CEO, and the company was called Network Systems. So, while we don't get any credit, local networking was also invented and or commercialized in the Twin Cities area by Network Systems Corporation. So about three years after this encounter with the venture capitalists, Jim came to me and said, "Can you help us as a faculty member because we have questions coming from the market we intend to serve that we thought you might independently answer. So, I'll come back to market in a second. And I said, "Well, what's the problem?" And he says, "Well, you know, it's expensive and it's very fast, and would be customers are unsure how it behaves as the traffic load increases. They wonder if they can use the bandwidth effectively or does its effectiveness collapse after some level of load. Well, my modeling stuff should work perfectly for this, right? It fits right in with what I've done earlier. So, I said, "What's the problem?" "Well, might it collapse under load?" "Why would one think it collapses?" "Because the only competing thing known to the world was Ethernet." Now, if you take Ethernet, in those days, it was a shared coaxel cable that you punched a big pin into to connect a node, and it had a ten megabit per second base rate, and it collapsed—collapsed is the wrong word. You could get the good of it up to about four-tenths of its base rate, I believe, I think. I may not remember this exactly, but you're going to get the idea. So, let's say four megabits of actual throughput on this thing was all you could get. And so, they were cheap little nodes, so you—they were PDP-11's and smaller machines and part computer science departments exchanging emails, papers, and so on. But" I'm not going to pay $40,000 for a box to connect to a Cray and have it fall over at four megabits or be limited". So, they said, "Can you figure out how to convince the world that this works?" That was the first thing we did. Whoops. You can't see this at all. This system was called HYPERchannel. (Visual of paper "Measurement and analysis of HYPERchannel Networks") Can you see it? It had a base rate of 50 megabits per second, expandable in increments of 50 megabits per second to 200 megabits per second, and was targeted to networking major computing platforms such as IBM mainframes, Cray computers, CDC 6600's etc.

Yost: Yes.

Franta: Ok. And so, this was our first piece of research on it. Now this was putting together a big test bed at Network Systems and a graduate student, the co-author is a graduate student named John Health who went off to be a faculty member over at the University of Maine, I believe. By the way, I have graduate students that are deceased, more than one. I had a very lovely—I'm sorry, I meander, don't I? A very lovely fellow, named Mark Bilodeau who was a Major in the army. They sent him to get his PhD and became a permanent instructor at West Point. We sent Christmas cards back and forth but then he disappeared, and then my wife had a friend who was an ex-West Point graduate, because I couldn't get into their systems, and he found out that the poor guy had died several years earlier. So, and that's just one instance. I'm a long survivor compared to almost anyone here. Remember I told you last time, whatever I say is true because all those who can refute it are gone. So, we did this piece of work on the HYPERchannel. Now, why is it that this is kind of significant? These people came from Control Data, so they were in the habit of building channels for computers, right? The 6600 had channels. So, this is a HYPERchannel in that its architecture resembled the organizational structure of a channel on a computer. The ethernet architecture better resembles a wireless situation, which is why it was compared to a wireless system called ALOHAnet. This later system interconnected stations on the Hawaiian Islands, hence its name, ALOHAnet, and it fell over at 30% of useful base bandwidth. And so, the world thought this is just a bigger, more expensive version of ethernet. And I remember that, at some point, while we were doing this, there was some symposium some place, nobody knew what the hell ethernet was or what the term networking might be, for anything. So, guess who the two guest speakers are for this gathering, Bob Metcalfe, founder of 3com, the inventor of ethernet, and me, and now at least—you know there still is an ethernet? The only thing that exists from the original ethernet is the packet size of 512-byte packets. Everything else is different. But it's still called Ethernet. So, he and I are giving lectures on networking circa 1974, or at least 1970 and something. And this was our first piece on why this one behaves differently because it does not collapse—and then we got more generally interested in access mechanisms for local networks, wireless or wired. And we did several things. So, I had three or four graduate students where we did deep mathematical analysis—invention of—and then analysis of how the things performed. And actually, later on, we turned Hyperchannel analysis into a theoretical, mathematically based analysis that proved mathematically what, more or less, what we had earlier proven experimentally. So, I was now connected with the networking people, and Jim Thornton, and solved their problem and used to go out and give these results on their behalf to various potential customers—well, who bought this thing? I told you that ethernet was purchased by schools, da, da, you know, this, small applications, low volume transfers. This thing, so ethernet was built using a 10-megabit per second capable cable, a piece of wire capable of 10 megabytes per second, action. This

Hyperchannel thing was 50 megabits per second, and you could have four cables, they called them trunks, that would get it up to 200 megabits per second. So, who buys this? Well, anybody that has big machines and insist on effectiveness. So, it's every bank in the world, it's every government agency in the world, it's the controller of the 6600 market, plus, because you didn't do a lot of banking with 6600s but with big IBM machines. So we got into the networking problem business with myself and graduate students, and that book is a result of being asked a simple question, namely, can you show the world why this doesn't fall over which we were able to pretty handily do. And the book was, you probably know better than I, published in '80 or '81. Something like that. And the only two real—

Yost: '81.

Franta: '81, and the only two commercially available systems in the world at that point were HYPERchannel, and ethernet and they were developed with very different markets in mind, each with very different cost point, bandwidth, and effectiveness needs .Today we would say that Hyperchannel addressed the data center networking market. I think that's the translation we would make today. We had shown its effectiveness under a wide variety of circumstances. There were three or four papers that came from an association with someone and being asked a simple question about effectiveness. The co-author on the networking book is the student who I was intending to visit this fall in Slovakia that we've called it off for a year. Now he was a professor at several places, including the University of Texas, Boston University, let's see, he was at the TECHNEON for a bit, and he's, he was in Italy for the last few years running some type of research institute, and I told him, recently, he should retire. He's now, he's only a short distance behind me in age. Time to retire. But that's how we got to the networking side and that book. Now along the way, remember I was a regular consultant to Honeywell, and we did several networking things there on behalf of government contracts. Some of that, a lot of that we were able to publish and we have book chapters, da, da, da, da, on some of that stuff but some of it we weren't able to publish. Then, circa 1980, there's one more book if you're interested. That's the book entitled *Formal Methods of Program Verification and Specification*. During this point in time, there were two people at least, and it seems Mats knows better than I, that proposed and expounded on very formal requirements for programs. I never met either one of them, but their names are E. W. (Edward) Dijkstra, out of ETH in the Netherlands, and C. A. R, (Tony) Hoare, out of a university in England, sorry [Oxxford University, and prior Queens Univ. in Belfast], just now I do not remember which one. And they were publishing all this stuff about formal methods, and as part of our Honeywell experience, you were getting asked to investigate things like program proof of correctness using programs called theorem provers. The intent, it seemed to me, was indeed to prove that every program deployed was correct, in that it did what it was supposed to do. And there were first level theorem improvers that existed that the government had access to. But this was partly because of the Honeywell experience of mine. So, the co-authors on that book are, Earl Boebert was the Honeywell researcher—

Yost: Yes, I interviewed him. I did that for a computer security project for NSF.

Franta: Yes, he's been my friend since, oh, God, '71, I guess, when I started going to Honeywell as a consultant. He was the one of the co-authors on this this book. The other co-authors are Helmut Berg, at the time a Honeywell researcher, and a graduate student, although not mine, named T. G. (Tom) Moher. And so we were getting asked a lot of things about proving programs correct because they, you know, there were people in research institutions from labs who thought they're going to have to do this. And so that book is a result of that exploration. And the basic conclusion started in the preface, that I wrote the for sure, because it says in there, in the preface, there's no intention, we do not believe that there is any intention that you're going to have to prove every program correct forever but if you know something about the subject, then it guides you to write better programs without attempting to prove them correct, and you avoid certain things and you favor certain things as a result of knowing something about what the proof means. And if you look at that book and you take a parallel program which we had to go with a lot in the Honeywell systems, they were all distributed systems in the military, it's an onerous task to prove their correctness, so that while Edward and Tony were theoretically correct, there was a practical problem, there existed a wall that became a barrier to doing this. I think the wall still exists, but I haven't asked Boebert lately. He's still doing things in that domain, and now lives in Albuquerque, and I haven't asked him. He sent me a birthday wish, my birthday was in May and he sends me a happy birthday and we exchange niceties and that's about it. So then, we're now up to about '80 or '81, and it's about time I took a leave from Departmental chores. So it became known that I was going to take a leave for a year, '80, and I start getting all these, well, you know, "Why don't you come, why don't you consider coming here?" both industrial and university. So, the end result of that story is that I had one endowed chair offer, named after an admiral, from Virginia Tech which I actually accepted, and then I was going to take leave. And I believe I had or was being considered for a second chair offer from the Naval Post Graduate School as Richard Hamming had been my host for my visit there, you know Richard Hamming, of the Hamming code.

Yost: Yes.

Franta: But I'm less certain. I don't have any paperwork that says I had an offer, but my recollection is that they were also going to offer me an endowed chair position, one named after a different admiral. So, having accepted the Virginia Tech offer, they were very happy and I was happy, and then for family reasons, which I won't go into, I actually didn't go. And so, then I really became available, you know. You're going to do this, do this, then why not... And so, you know, for some reasons, known and unknown reasons, I decided, I had published three books, four or five book chapters, and more than 150 papers, maybe it's time for another adventure. So, I went out and began the phase of my life you're not interested in as an executive something in technology companies, that included venture capitalist, corporate treasurer, and after two or three such short adventures, I finally arrived at Network Systems where I spent 10 years. I had several roles between vice president, CTO, and finally general manager in Jim Thornton's office at the end of my career there. We sold Network Systems to StorageTek. But those were 10 to 15 very interesting years, and there was a small host of pretty capable people like my cadre of graduate students, and I learned as an aside. I learned how different it was "out there". Then using the constructs of the book on simulation for a customer, I developed an enormous model of network behavior that you could get very accurate predictions of what was going to happen with the real system, if you bought it and then started it. And that ran on a 6600 under that Simula construct. And so now I'm at Network Systems. It's 1987, and that particular customer asked me if I could do some additional work. I'm working for Network Systems, and I still had the code, but I didn't have access to the 6600 anymore, well, I suppose I could've paid for it. So, one of these bright lads that I had in one of my advanced development labs came in and said, "Well, why don't we just run it on the SUN workstation?" and I said, "Well, you know, man, it takes 20 minutes run it on the 6600, you know. It's going to take all day." "No, no." he says, "It's going to be just fine." So, we got the card deck, which I had. He got it all transferred to something to get it into the SUN workstation. It ran in under 10 minutes on the SUN workstation, which was the first time it really sank in that workstations had now taken over from supercomputers, for many reasonably "intense" applications. Anyway, I spent about 10 years at Network Systems inventing more things and making more friends and traveling the world. I gave lectures everywhere on networking in those first years. If you really want a funny aside, I, my first speech in Japan, I have a room full of maybe 200 people. And the deal was I had to write down my speech, which I'd never done in my life. I mean, I had notecards that said, here talk about this or this, but I never wrote it out. So, I had to write this whole first speech out. And then there was a young lady who would, when I went there, she had it all marked up and she'd ask me questions about various pieces of it. And the deal was that I would do a couple sentences of my speech and then I was to pause, and she'd translate it into Japanese. So, we did this a couple times. I do two minutes, or two sentences, or three sentences, about a minute and a half, two minutes, and she did five minutes, and we did that back and forth a couple of times. And I said in English to the crowd, "I don't know for sure, but I think she finished my speech," and they all laughed hysterically. There was no one in the room that didn't understand English. So, there's my career at the University. And along the way, we did some other things and all kinds of other stuff that we hadn't broached. It was a pretty interesting time with nice people, great things to do, everything was new, and as they said, I was, you know, I was very fortunate in what I got to do, I got to do the math that I liked, I got to do the modeling and simulation stuff that I liked, I got to do all the networking stuff that I liked. It was—I had a nice career, I thought. I enjoyed it.

Yost: Good.

Franta: And that's important.

Yost: It is. I haven't asked you about the different leaders of the department, starting with Marvin Stein and then J. Ben Rosen. Can you talk a little bit about the leadership of the department and what these individuals brought to the job?

Franta: Well, Marvin brought the founding—Marvin was a sweetheart in many ways. He was in many ways a fighter, but he was a benign fighter. He was very nice to everybody, especially me. So, his, he pretty much let everybody do what they wanted. And it worked out, and when you got eleven people, it worked out just fine. I don't know how well it would work for Mats with 48 or 50 people because I never lived in a department with a faculty of 50. I had well over 100 employees at one point, you know, but anyway, but Marvin didn't dictate direction to anybody. He came to us with the problems of the department, you know, this funding problem or the Dean's office wants this, or they would like that, could we do this, how do we offer more versions of this class, you know, it was practical. He was in no way domineering or dominating. He was a sweet man. Ben Rosen was a lovely individual as well. He had a slightly—I don't know how to characterize the difference between them. Ben Rosen was a little bit more formal in his relationships with people, I thought anyway. I just thought of that, that was one of describing it, making the distinction. Marvin was quite informal. Didn't come to you in any particular way. Ben was a little more formal, wanted things done in certain ways. Nothing to be offensive about, it's just a different management style, and everyone has their own management style. A little bit different. And then when—I don't actually remember why Ben left. He left for some reason. Maybe you know. I don't know.

Yost: I don't.

Franta: When he left and then Kurt Maly was acting department head and, you know, I had a feeling no one paid attention to him, and I was so busy doing my thing, I didn't see much of him. I'd go talk to him once in a while but he, he didn't, you know, he's the guy I said you should go talk to. He's at Old Dominion.

Yost: Right.

Franta: He's a nice fellow too. He and I did several pieces of work together, published in ACM, not IEEE. But I always had this feeling he was caretaker department head, and that others considered him unimportant. Anyway, true or false that was my impression. I thought he was kind of overlooked for much of it. And then I don't have anyone after that. I'm still partly responsible because I was head of the recruiting committee for some of the remaining faculty, David Du, Maria Gini and Dan Boley and a couple that have—are now deceased. But there's still some remnants of some of my work there. It was a fun time. It was a really fun time. Nobody asked me but if you want this question covered, I will be happy to do it. Somebody had wanted to ask me in the Emily video about what am I interested in today, and what do I think of computer science today, if you're interested—

Yost: Yes, please.

Franta: A couple of remarks. Mats worries about whether CS is a science or an engineering field, and I think it's an interesting question, it's kind of like, the question is math discovered or invented. It's both, I think. Somethings you discover, somethings you invent. But you know there's a formal side and there's the complexity side. Then there's the engineering software side. So, I think it's both. Today, here's what interests me today in the field. Doesn't mean something else wouldn't if I was more attuned to it, but, a friend of mine asked me several months ago when we were, we used to have Zoom cocktail parties on Saturday nights when we didn't or couldn't go anywhere, he asked me one night if could I explain quantum computing. I thought about it for 15 seconds and said, "Not effectively, but I'll get back to you." Well, it's now the middle of July, it's months later and I've yet to get back to him again, but I'm looking into it, I mean, at a pretty deep level, and it's again reinforced my notion that the more mathematics you know the better, doesn't matter what field you're in, but especially in computer science you're better off the more math you know. Quantum computing is so completely driven by linear algebra constructs that I'm able to make sense of it, after a bit of review of the subject. I haven't finished my delve into quantum computing yet' but I'd be completely lost if I didn't have the background in mathematics and it's a hobby so I can proceed at my own pace. So, I think that all the math you have as a background for computer science, the better off you are. I always thought that it provided some basic model for how you think about problems, issues, and material. Remember, I said asking the right question is the all-important part, I think it helps with that if nothing else but, in this case, going into quantum computing it provided all the machinery I needed. It's mainly linear algebra end to end except a bit more abstruse when you consider posterior states and or collapse of quantum states following a measurement and or entangled states. In any case its like no model of computing I ever considered during my active days, and as a model of computation and programming it is lovely and worthy of study even if we never are able to build a machine with a sufficient number of "qubits" to be truly useful. So that's one thought. The other one concerns AI. There's a fellow, I don't know even how well-known he is, named Pedro Domingos, from the University in Washington, and he's written this, book about the Master Algorithm of AI, where you can reconstruct with one algorithm all the knowledge of human kind in every field because you now have the master AI algorithm. The search for said algorithm is a serious pursuit. And I just can't help thinking that he's going to run into a Kurt Gödel-type barrier somewhere along the line in that quest if you know what Kurt Gödel did.

Yost: Yes.

Franta: I just, I'm just having trouble with that as a real outcome. Secondly, I was happy to be invited to the 50th anniversary celebration at Alumni Hall, as I think it's called. And they had a keynote speaker from Microsoft, and she scared me a little bit because it seemed, but I could be a wrong in my impression, but I'm not the only one who had it, I sensed that several of the faculty members had it, had a similar feeling that from inside a university setting, the major tech players who can collect all the data on people behavior are not sharing the data that they collect with anybody, and they are also not sharing the algorithms used to influence people behavior. To me, they're figuring out how to manipulate people, and that's scary to me in the end. I mean, that's where it's going to end up, but the prospect was just, something rang in my ear and it concerned me. AI concerns me a little bit, and that's my distortion of the remembrance.

Yost: Yes, I don't know if you know the book by Shoshana Zuboff on surveillance capitalism, but she writes about the degree of data that's being collected on people, on analytics, and how it's escalated over time and the way it's being used and characterizes our age as an age of surveillance capitalism.

Franta: You know, you can make this off the record if you want, but when Trump and others in the government you got to be careful about some of these Chinese companies, they're collecting data on all of you, you don't know if that's true or not but you have to suspect that somebody knows something that I don't know anything about, and if they are, then it's a bit scary because you can manipulate, I mean, that's what this is for, to manipulate people. And, I don't know, that just bothers me. So, there's my two-hours, almost our two hours. I had a fun time doing this. Thank you. If this goes into your little library, will you let me know, because then I'll share it with others.

Yost: Yes, definitely. So, what will happen is that the Computer Science and Engineering Department will take the video, we'll transcribe it at CBI and we'll give you a chance to edit the transcription and then the transcription will go online.

Franta: Ok.

Yost: And that's what historians tend to use, and then the video, it's up to Mats and his department what they do with it. But I'm sure if they are putting parts of it up on the web that he'll be in touch with you about doing that.

Franta: You know, as a complete aside, you always think of things, having gotten into mathematics, as I said, not part of a plan but a response to what appeared to be an opportunity. Having gotten into computer science was not part of a plan but a response to what appeared as an opportunity. My life was not well planned. It's been all about leaping onto presented opportunities. We happened to have had, a very robust Math Department, I don't remember what we were ranked at but I recall, or imagined, a small number, fourth or eight or somewhere like that, when I started my mathematics training. You know, we had quite a faculty here, built up over the war years perhaps, maybe because you couldn't bomb the place during World War II. You couldn't get an airplane in this far, and for that reason, as I recall, maybe falsely, for that reason naval guns and bomb sites were built here. My Father was part of a crew who prepared the Dolittle bombers here, so he said. Did you know that?

Yost: No, I didn't.

Franta: Maybe I'm wrong. In any case, we had a really lovely math department. I remember Hans Weinberger and a guy named Bill Ellis and Bill Munro. Bill Munro, you know, I believe did work on this navigation system based on hyperbolas instead of other mechanisms. LORAN, I think it's called, LORAN. In any case it was an exciting place for both computing, and networking.

Yost: And this interview shows, with the industry here, it created opportunities for both in networking, and broadly in computing and computer science.

Franta: As I said, I had enough funding from industry that I hardly made use of other stuff. Sometimes papers said it was funded by a government agency but that was generally through my work done with Honeywell, the Honeywell group and the others. It's—let me see, there were a couple of people. Grace Hopper was one of them that you don't seem to have any video from her. Peter Denning you do, Neil Lincoln you do. Neil Lincoln is another fellow, was key to development of the STAR machine, a vector processor also developed by Control Data. I lived two miles from where they built that in the '70s. Jim Thornton you do have something on, right, I think. And, by the way, this is capricious, somebody said to me, "Why did you leave the University? Nobody does." My tongue in cheek response was, well, there were two gurus, of perhaps several, of note in '80, that cited me in their books. One was Donald Knuth and one was Leonard Kleinrock. Leonard Kleinrock. And I said, "Well, I have a citation in Knuth's second volume on semi-numerical algorithms, and I have a citation in Kleinrock's book on queuing theory so what more can I do?"

Yost: Well, congratulations on an amazing career, and I'm so grateful to you for taking the time to chat with me this afternoon.

Franta: I'm happy to do it, and I, you know, you don't get to talk about all the nice people along the way like Jim Thornton and, of course, Peter Patton, you know who he was.

Yost: Yes.

Franta: And on and on it goes. Grace Hopper was...I got to tell you the Grace Hopper story. It was—do we have time?

Yost: Sure.

Franta: So, as I told you, I had her come in for this course on computers and society, when she gave the talk with the eleven-inch pieces of wire. So, I pick her up at the airport, and these were the days of course when you could go to the gate to meet your party. Now, she was an admiral in the Navy, active duty. So, I pick her up at the Minneapolis airport, and she looked—find a picture of Grace Hopper in the Admiral suite—she had her full uniform on. Unfortunately, she'd fallen down the day before she came to us on the ice in Washington, D.C. and badly bruised the side of her face, cut it in a couple places and her eye was black and very swollen. And all I can remember when we're walking through the airport, taking her out to my car, is that when you are walking down the hall with a 70 year old woman in a sailor suit who appears to have been recently beaten, everybody in the corridor looks at you with a suspicious eye That's my story as I tell it. Peter Denning is another a nice person. There were so many. There were so many nice people who passed through my life and, unfortunately, as I was thinking about this for you, I don't know where many of these people are. I have no clue, too bad. It's a nice thing that you called on me before it became impossible to do. It's, we are dated now—ha—stuff that I did is no longer very exciting. It was then more so.

Yost: Ok. Thank you, Bill.

Franta: So, you'll be in touch, letting me know what I'm supposed to do next.

Yost: You bet. Thanks again, Bill. Bye.

Franta: Bye.