An interview with
Albert M. Erisman

Conducted by Thomas Haigh
On
6-7 July, 2005
Bellevue, Washington

Interview conducted by the Society for Industrial and Applied Mathematics, as part of grant # DE-FG02-01ER25547 awarded by the US Department of Energy

Transcript and original tapes donated to the Computer History Museum by the Society for Industrial and Applied Mathematics

ABSTRACT

Albert M. Erisman grew up in the suburbs of Chicago. He earned a bachelor's degree in mathematics from Northern Illinois University in 1962, during which he developed an interest in journalism. On graduation he worked for two years as a teacher, before returning to school at Iowa State University where he received his applied mathematics Ph.D. in 1969. A course in numerical linear algebra with Bob Lambert kindled his interest in numerical analysis and led to thesis work in the area. Erisman then went to work for Boeing in Seattle, where he remained until retiring in 2001. He began as part of a small group of mathematicians, providing internal consulting services to projects in need of mathematical expertise. He became closely involved with Boeing's internal library of scientific routines, which under the name BCSLIB was also used by external clients as part of Boeing Computer Services' timesharing service and later software sales offering. Erisman rose within Boeing Computer Services during the 1970s, managing its mathematical and modeling services. He discusses BCS and its business strategy, the role of applied mathematics within Boeing, and the development and function of BCSLIB. Erisman also explains a shift during the early-1980s toward use of Cray systems, and the challenges involved in optimizing a library for vector systems. Boeing's success in this area led to work producing computational kernels for computer vendors. Erisman's responsibilities continued to broaden, and by the 1990s he was Directory of Technology for Boeing, with broad responsibilities for IT and mathematical research. He discusses the contributions of scientific and technical computing to Boeing's success, including the FlyThru technology used in the design of the Boeing 777 jet. Erisman's main personal contribution to research in mathematical software came in the area of sparse matrices, including his long-term collaboration with John Reid and Iain S. Duff resulting in numerous papers and the book *Direct Methods for Sparse Matrices.* This interest grew out of applied problems faced by Boeing early in his career, and throughout the interview Erisman stresses the importance of a deep understanding of context and problem areas in the design of mathematical software. Since leaving Boeing, Erisman has devoted himself to a new career exploring the intersection of business ethics and technological change. He has been traveling widely as a speaker and consultant, and directing the Institute for Business, Technology and Ethics and its publication, *Ethix* magazine.

HAIGH: I wonder if we could start by talking a bit about your family background and early life.

ERISMAN: My father was an engineer. He was Chief Engineer for a company called Link Belt, which makes construction equipment, now a part of FMC. My grandfather had been Chief Engineer of the same company before then. My uncle was an engineer for International Harvester so I came from an engineering background. My mother did not work outside the home. I have an older brother and a younger brother. My older brother went to Northwestern and got an engineering degree. So it was pretty much an engineering family, and I decided I wasn't that interested in being an engineer. I loved mathematics, and I was involved in accelerated mathematics programs in high school, which was Oak Park, in a suburb of Chicago. I ended up going through calculus in high school and I just enjoyed it a lot. So when I went to college and picked math as a major I wasn't really quite sure why I did, except that it was fun.

Along that period of time was the Sputnik event, in 1957, when Russia launched its satellite. There was a lot of interest in mathematics. I saved for many years a paper I got in high school about careers in mathematics, and it said if you really worked hard you could earn up to $5,000 as a mathematician and maybe even more if you got a Ph.D.. So when I went off to Northern Illinois University as an undergrad, I majored in math—that was an obvious choice. But I got interested in other things as well. I joined the newspaper staff. I was sports editor and my senior year I was editor of the college newspaper, so I picked up a minor in journalism, a minor in physical education, and a minor in education. And I was really not sure at all what I wanted to do when I graduated.

HAIGH: Let me just take back over that ground with a couple of follow up questions. It's interesting that you say that you in some ways reacted against the family tradition of engineering. So what was it about engineering that you didn't like?

ERISMAN: I think it was the long hours that I saw my father putting in, which I ultimately put in throughout my whole life. I think that engineering at that time seemed to be an isolated kind of thing, and I always loved collaboration, so I guess that was part of it. I guess the other thing is that I don't consider myself particularly good at gadgets; it's just never been something that has fundamentally attracted me. I like tools and I like technology, I love what it does, but I'm not that interested in exploring all the different features and all the devices. I think most engineers have kind of a love of that and it just wasn't a part of who I was.

HAIGH: Did you have any of those hands-on technical hobbies like ham radio?

ERISMAN: I built a few radios, yes. My older brother is the one that led that though, and I did it because he did it; it wasn't that that was as interesting to me. I was more interested in baseball. Working I had paper routes and shoveled snow and did lawns and all that kind of stuff. It certainly isn't the stereotype from the colleagues and friends I've talked to.

HAIGH: As your interest in mathematics developed over your high school career, did you find yourself interested in applied mathematics, or at that point was your education confined to pure mathematics?

ERISMAN: At that point it was basically pure mathematics; the applied mathematics came much later.

HAIGH: How would you describe the quality of the high school mathematics education that you received?

ERISMAN: I was fortunate—it was outstanding. Oak Park High School was one of the top schools in the country, and I had an opportunity to participate in the accelerated math program there and had some good teachers. I was very fortunate.

HAIGH: Can you remember the first time that you became aware of the existence of computers?

ERISMAN: That was an interesting question and I have been thinking about that. I suppose trips to the Museum of Science and Industry which we used to take when I was in Chicago made me aware of computers sometime in high school. But I graduated in high school in 1958 and that was really before computing had made the inroads into society. I certainly didn't do any computing even as an undergraduate. I think during the late '50s, though, as I finished my high school and started college, people became more aware of the environment of computing because of the Sputnik and the rocket programs and the space programs. But I didn't have hands-on experience at that time.

HAIGH: So nothing more than general background awareness?

ERISMAN: Right.

HAIGH: Now you have talked a little bit already about your undergraduate education at Northern Illinois University. Why did you choose that university?

ERISMAN: I got a scholarship. I wanted to get away from home and live at the university. I wanted a fairly small school. It was fairly small when I started there, though it was kind of large when I graduated—it grew very rapidly during that period. I wanted a small school where I could try some things out and experience some things. I have to admit it wasn't a great deal of thought.

HAIGH: How would you describe the school's strengths and weaknesses at that point?

ERISMAN: Well I think mathematics at Northern Illinois University was going through a major transition. It had been a teacher training school, and the kind of courses that they offered were pretty limited. I took every course they offered including some graduate courses in my time there. But in my junior year a professor came from the University of Pittsburgh who was probably stronger than anyone else on the staff, and he provided some leadership for me and I had some good courses from him.

HAIGH: Do you remember his name?

ERISMAN: John Christiano. There was another professor there who was a well trained mathematician, Jim Beach. Between the two of them I took all of my upper-division math courses from them, and I got to know both of them fairly well. But it was also true that by my junior year while I was committed to my degree in mathematics, I wasn't that interested in mathematics anymore. That's when the study of journalism, I became editor of the college paper, I thought I might want to be a coach, so I was really experimenting with other things. I had a wonderful course in art history. So it was those kinds of things that made me think about education much more broadly than mathematics.

HAIGH: So it sounds in some ways that's the opposite of the normal process, you went in with a well-defined interest and broadened it as you went along.

ERISMAN: Right. I admit by the time I graduated I had not really thought at all about a career in mathematics.

HAIGH: Would you say that you were particularly good at mathematics at this stage?

ERISMAN: I was pretty good by the standards at that university. I remember one course I took in the history of mathematics (which I am fascinated by now, I have read quite a number of books recently), but I was not interested in the class at all. It came to the final exam and the professor said, "You do not have to take the final exam if you are getting a C or better and you can take the grade you have." He gave my grade as a D. So I studied for the final exam, I did really well on it; I think I got a B in the class. He told me later, he said, "You weren't really getting a D, but you hadn't done any work all semester and I wanted you to do some work." I thanked him for it; it was a good kick in the rear end.

HAIGH: Did you have any interest in business at this point in your life?

ERISMAN: None. I really thought that I would be a journalist. I loved my role as editor of the newspaper and sports editor, I loved the travel, the interviews, the writing, and that's really what I thought I would do.

HAIGH: Now as graduation neared, it appears what you actually did was become an eighth grade teacher and coach in the Kane County School District, Illinois. How did that come about?

ERISMAN: Well it came about because of my associate editor on the college newspaper. We developed a very close relationship and we married in August after I graduated; she had one year of school to go. So the objective was to live in DeKalb, Illinois so she could finish her senior year. So my job search was limited to places close by. I had a degree in math and I was qualified as a teacher, I had done my student teaching, and so it was natural to look for a position in teaching. I found this position as an eighth grade teacher and basketball and track coach at this consolidated country school district about ten miles from DeKalb. I did that really just to have a job nearby. I got paid $5,000 a year and a $200 bonus for coaching. That first year is what really changed my outlook.

HAIGH: How did it change it?

ERISMAN: I was one of the few people in the school district that was actually trained as a mathematician. This was at the time when they were putting modern math in the curriculum, and as mathematician I really did understand what they were trying to do with modern math. I had an accelerated class. I was teaching four eighth grade sections, but one was the advanced section, the kids were really at the top of the class, and I taught math to that group. I had four boys in that class that asked questions, that were curious, that challenged me, and I give them credit for totally rekindling my interest in mathematics.

I remember my favorite event. I was introducing to the students the idea that the product of two negative numbers is a positive and this got everyone with quizzical looks on their faces except one boy. He raised his hand and he said, “Mr. Erisman, if three times three is nine and negative three times negative three is nine, then the square root of nine must be plus or minus three and not just three. Is that right?” I said, “Wow.” From that point I was back in the library, I was reading things about infinity and talking to my students. For this group of four boys, I developed a curriculum just for them and they were working in the corner of this accelerated class and doing a great job and really challenging my own interest. So at that point I decided to go back to Northern Illinois University and get a master’s degree. I started taking classes at night and in the summer, my first summer. My grades were much better because my attitude was better. I did well. Midway through my second year of teaching eighth grade I realized that I was very interested in mathematics and I didn’t really have the training I needed, and I thought I would like to go to graduate school, and not just night time and summers, but full time. So I convinced my wife we should find a graduate school and go get a master’s in math; that was the goal at the time.

HAIGH: So at that point did you have the idea that you would be returning to school teaching?

ERISMAN: My goal at that time was if I got a master’s degree I would return to high school teaching, and I had the vague idea that I might want to get a Ph.D. and if I did I would be to teach at a university. That was the goal.

HAIGH: Other than rekindling your interest in mathematics, do you think that this teaching experience had any other impact on the way you saw yourself or how your career developed subsequently?

ERISMAN: I have always enjoyed the idea of making presentations in a way that can clarify a subject to non-experts, and whether that’s teaching or some of the presentations that I did at Boeing, it’s really all the same.

HAIGH: With the dates I have from your resume, it says that you were teaching from 1962 until 1964.

ERISMAN: Right, that was two years, the two school years.

HAIGH: Then you received your master’s degree in applied mathematics from Iowa State in 1967.

ERISMAN: Right.

HAIGH: So had you moved there in 1966?

ERISMAN: No, I moved there in August of 1964 and started school in September of ’64. I was enrolled in a mathematics degree program. The reason I went to Iowa State, quite frankly my grades were up and down and all over the place with all of the fooling around I had done as an undergraduate. So I went to Dr. Beach, who was one of the two math professors that was really a help at Northern Illinois, and he said, “I have some good friends at Iowa State, and I think you can do well there and I will recommend you for a scholarship.” The reason I went to Iowa State is because he recommended me and I got a scholarship, and so I went. It was nothing more profound than that. In 1964 I started on the masters program and I soon realized that I really liked the study of mathematics, so I decided to just go on for Ph.D.. I had actually started the work on the Ph.D. dissertation and I wrote part of it for a masters degree and kind of formally finished that just so I would have the degree in 1967, but it was January of ’69 when I got my Ph.D.. So it was just kind of a stop along the way.

HAIGH: So quite soon after you arrived you made the decision to go for the Ph.D.?

ERISMAN: Right.

HAIGH: What kinds of things were covered in the graduate curriculum at Iowa State in those days?

ERISMAN: Well, my first year I remember taking abstract algebra and real analysis. I added a course I think in number theory. Then in the summer of ’65 I took a class in numerical linear algebra from the person who became my major professor, Bob Lambert. I really fell in love with numerical linear algebra at that point. I had not had even a course in matrix theory, but this was something that intrigued me and excited me, and it started me thinking about the research topic I wanted to follow and so on. I think my first quarter at graduate school I got B’s in both of my classes, and after that I ended up in the rest of my Ph.D. program getting maybe two more B’s (or one more, something like that). It was just a question of getting an attitude straight and understanding what it took and getting motivated and excited, so it was great. I loved my time at Iowa State.

HAIGH: Were there any other courses in numerical analysis?

ERISMAN: Lots of them. In fact, one of the opportunities that I had as a graduate student was to do a lot of teaching through my fellowship. So I started by teaching calculus, and that’s when I really learned calculus, incidentally. Then I taught advanced engineering mathematic, I taught undergraduate numerical analysis. So I got an opportunity to really teach; it wasn’t just assisting another professor. We had the class ourselves for the undergraduate courses and I loved it.

HAIGH: Was the use of electronic computers integrated into the graduate curriculum at that point?

ERISMAN: I wouldn’t say that it was completely integrated, but I started doing my actual computing during this time. If I could step back just for a moment in the eighth grade when I was teaching, the advanced modern math program really introduced the idea of computing. We started experimenting theoretically. In 1963 Time Life put out a book on mathematics which really captured a lot of the ideas and talked a lot about computing, so that’s when I really got interested in that. I started using computers in graduate school. I kind of learned Fortran on my own as a graduate student, and in my dissertation I ended up doing a lot of computation, so I was probably using the computer center at Iowa State by my second year there.

HAIGH: So what kind of computer facilities did they have?

ERISMAN: It was a computer center where you would take your deck of cards and you would hand it to someone, and if you were really lucky you would get it back in a few hours; if you weren't quite so lucky you would get it the next day. I remember talking with one of my colleagues saying, "Wouldn't it be great if they could put a flag on the building that would say the computer was down so we wouldn't have to walk all the way over there to see if our job was ready." So there was no concept of kind of remote connections to the computer. It was one of the early IBM models, and I don't remember which model it was.

HAIGH: So you never personally would have touched the computer?

ERISMAN: Never. Not until I got to Boeing did I actually touch the computer. I do remember a graduate's assistant-- I later had a fellowship through the Ames Laboratory which was on campus there at Iowa State. I was doing research and there was some interns that came in for the summer, and I had one guy that was really sharp in computing. He decided he would play a trick on me, so he copied an overflow message and just printed it out on my sheet just to see what I would do. Well it turned out my job failed because when this error message printed ten times, that was the key for the computer to kick it off. So it was reading the output file as opposed to looking for an actual overflow. I remember thinking that was a really stupid way to deal with an error message.

HAIGH: Can you remember your first programming experience, how you reacted personally to the process of writing a program?

ERISMAN: I remember it as being very magical that you could create an idea and you could kind of write down some steps and then you could do it. I also remember the level of frustration though, because I tend to be a conceptual person. I finally acknowledged that after many years. And you have to be very, very precise when you do computing, they do exactly what you tell them to do and not what you wanted them to do. So the unforgiving nature of the computer was a very early source of frustration.

HAIGH: So when you took a course like the one you mentioned in numerical linear algebra, would that have included programming assignments?

ERISMAN: It may have; I just don't remember. I do remember some of the theory we developed in looking at computational methods, and we probably did some programming then. That was probably the first, but I just don't remember for sure.

HAIGH: Perhaps you could talk some more about your dissertation project: the advisor, the topic.

ERISMAN: Okay. I ended up doing something in numerical linear algebra. My professor had an idea which I carried out research on for both my master's and my Ph.D.. Ultimately I think it was an interesting idea that didn't really go anywhere, but the idea was instead of performing the calculations of eigenvalues for a large matrix by going through a series of transformations, all of which introduced round-off error, that there should be some way you can work with the original matrix and continue to compute functions of that original matrix. We ended up building some algorithms that did this. They involved kind of inventing some algorithms in mathematical optimization to carry out the calculations, and I wasn't as deeply schooled in that as I should have been. So ultimately it didn't really take me where I wanted to go, except it was a dissertation. But I never really turned that into software, I never really turned that into papers and so on.

HAIGH: Do you remember what the title was?

ERISMAN: "*An algorithm for eigenvectors of non-hermitian matrices*". My colleague, another graduate student, worked in a parallel way on the symmetric matrix problem, the Hermitian matrix problem, and that was a more tractable problem. I ran into all sorts of issues on the non-symmetric part, but it was fun. I haven't looked at the dissertation in years so I don't remember precisely.

HAIGH: So you have mentioned that the content of the dissertation itself didn't lead on to anything. Did any of the relationships that you established in graduate school play any part in your later career?

ERISMAN: Very much so. In 1967 or 1968 my professor introduced me to SIAM, and we went to local chapter meeting of SIAM in Iowa City, which was about 70 or 80 miles away. So we rode in his car and went to this meeting. One of the speakers at that meeting was a fellow named Bob Brayton from IBM. Bob introduced an idea, which at the time I thought was a little silly: a sparse matrix computation. He showed why taking advantage of the zeros in a system of equations was a really good thing. I got the idea, though it didn't really mean anything at the time. But it became a really big thing after I got to Boeing. So that relationship with Bob Brayton was important. Other influential relationships included my professor and Richard Sincovec, the fellow that I shared an office with. He also worked with mathematical software, and we became lifelong friends and colleagues.

HAIGH: So is there anything else you think we should discuss about the Iowa State experience before we move on to Boeing?

ERISMAN: No, I think that probably captures it.

HAIGH: All right. I think you said earlier that your idea had been if you had got the Ph.D. you would probably want to work in teaching at a university. So as you approached graduation how did you start to plan for the future?

ERISMAN: I made two decisions: one is that I would not interview for any university position, and the second was that I did not want to live in the Midwest anymore. The first one was made on the basis that I felt to be an effective university professor I needed to know more context of the mathematics, and I decided it would be a really good idea to go to industry for five years and find out why anyone cares about the mathematics that I had been working on so many years, to establish context which would then be useful in my teaching. So I interviewed at IBM in New York, at Pratt-Whitney in Connecticut, at Union Oil in Southern California, and at Boeing in Seattle. So East coast, West coast, and all industry.

HAIGH: So then the second part of that, why were you so keen on getting out of the Midwest?

ERISMAN: I just had lived there all my life and thought I would like to live in another part of the country. My wife and I had this thought at the time that we could live one place for a few years and live another place. So in 1969 when we came to Seattle we thought we would be here for a few years, I would be at Boeing for a few years, and we would go away to teach, number one, and we would go away to some other part of the country.

HAIGH: So to follow up on that point you made about the need to understand the context of the mathematics, do you think that is something that is specifically true about these particular kinds of applied mathematics as opposed to mathematics as a whole?

ERISMAN: Well, I have a lot of colleagues who have been at universities all their lives, and they have had wonderful careers. That just doesn't reflect me. I think whether it is in applied mathematics or in physics or in chemistry or in anything, the idea of understanding a context is really important. Recently I have gotten to know a brain scientist and understand how the brain really works and how people learn. One of his brain rules is that people understand meaning before detail. If you can get a context which establishes why you want to learn something, that creates a much better learning environment. It's something that I intuitively believed then and I more scientifically believe it now. I would certainly encourage others to really get their hands on real problems first and then go off and teach with a context. It's just something I strongly believe in.

HAIGH: To jump ahead slightly, obviously over the last 30 years you have been in contact with many, many of the people who have been involved in producing mathematical software libraries. So do you feel that this community as a whole has risen successfully to the challenge of incorporating the context into their work?

ERISMAN: Not completely. My belief is that the reason you want mathematical software is to solve a problem, so the question is what is the nature of that problem that needs to be solved? Sometimes as a mathematician you can see that something would be an interesting problem, but the question is: is it relevant to something. Or maybe as a mathematician you won't see a problem class at all. That was one of my eye-opening experiences over the time at Boeing. I found that there is no real mathematical construct that would differentiate a matrix arising from three or four dimensional applications than from two or one dimensional applications, and yet it makes a big difference in the way the problem is solved. There is no mathematical construct that deals with a complex symmetric matrix. A complex Hermitian matrix is one that has well known properties, but not complex symmetric, but they arise in practice. Understanding and taking advantage of the problem characteristics from a context does create a collection of interesting problems that I enjoyed working on and that I felt needed to be thought about as a part of a mathematical software library. They kind of capture the problems that really need to be solved in reusable form, and that's what kind of drove our own library development, and we will get into more of that.

HAIGH: So you are saying there are distinctions there between particular kinds of problems that might be important to people who actually have problems to solve, but wouldn't be traditionally interesting to mathematicians?

ERISMAN: That is correct. In fact even the whole sparse matrix field kind of arose from an applications point of view as opposed to a mathematical point of view. One of the historical figures in sparse matrix work is a fellow named Bill Tinney from Bonneville Power Administration, Portland. I got to know Bill quite well over the course of many years. Bill had come up with this idea of taking advantages of the zeros in a matrix and solving a sparse matrix problem and had shared it at Stanford University in the '50s. They said, "Well this is just Ax= B. We have already solved that problem, it's not of interest." So it was kind of interesting to see that early on there was some context of problems that was not recognized at a mathematical level.

HAIGH: So we will return to that theme later then. Now you have mentioned that you had interviewed with a number of industrial employers. How did you make the decision then to go with Boeing?

ERISMAN: I think in the end it was down to Unocal in Southern California and Boeing just because of my interest at the interview and the places to live, California and Seattle. Unocal was a wonderful opportunity; I guess ultimately we decided not to do that because we liked the idea of the Seattle area better than the Southern California area. So the environment and the geography was a strong influence in the final decision. The offers that I had were about the same about of money, facilities were much nicer at Unocal, the freedom and flexibility to do what I want would have been greater at Unocal, but ultimately I chose Boeing because it seemed to me that Boeing had a very, very difficult set of problems to solve, and that was challenging to me.

HAIGH: So how would you describe Boeing's position at this point in time?

ERISMAN: At this point in time? Boeing has been through an incredible amount of transition. They have always dealt with a difficult marketplace. When you make huge products that are very capital intensive, the market will have cycles, and so Boeing has always gone through cycles. In fact the first one happened shortly after I got to Boeing. I joined in January of 1969, there were 105,000 people in the company. I was told that the company was pretty stable because they have a mixture of military contracts and commercial airplane contracts and the markets tended to offset each other, so they create a lot of stability. Two years later Boeing had 35,000 employees, factor of three reduction in two years. I went to my boss one day and said, "This is a pretty scary environment right now. Should I be looking around? How am I doing?" He said, "Well, you are doing fine, you will be here after I'm gone." I said, "Well, that's great," and two weeks later he was gone! It was a tough environment then, but there you kind of understood it from a market point of view.

It was even tougher later, 1996, '97, when Boeing started going through all these mergers, because mergers meant that you no longer controlled your own corporate culture. I think Boeing is still recovering from the mergers of the late 1990s, and it certainly made a difficult environment. That was I would say a different point in history than the first, although both are transitions that the company went through.

HAIGH: So what was the downturn around the end of the 1960s associated with?

ERISMAN: Well, there was a difficult economic time. I think Boeing went a whole year without a single commercial airplane order. Boeing at the time was under contract to NASA to build the supersonic transport, and the plug was pulled on that. The whole market in the military airplane area was down as well because of the economic climate, and all of the military money going into

Vietnam. It was just a confluence of really horrible things. That was the time when there was the famous sign put up in Seattle, "Will the last one out please turn out the lights."

HAIGH: How would you say you experienced the culture of Boeing as opposed to the academic environment that you had been in previously?

ERISMAN: I would say that we had a very collegial environment. I joined kind of a pseudo R&D organization of the time. It later became a R&D organization, but at the time it was a group of advanced degree mathematicians who were out trying to help solve Boeing company problems all around the company. We had very little research funding so we were always scrambling for funds, but we had a collegial environment to work on projects and do interesting things. It was exactly what I was looking for—the idea of finding a context of hard problems to solve and having the freedom to apply the skills that I had gained to solve them.

HAIGH: So essentially that was corporate stuff you had been doing internal consulting?

ERISMAN: Internal consulting, right.

HAIGH: Did that kind of activity have a long tradition within Boeing?

ERISMAN: It had started sometime in the mid to early 1960s. In fact, it had started in two separate places. At the time I think it was called the Pilotless Aircraft Division, which was the military side, and the commercial side. There was a mathematics staff in both of them. I joined the military side and we worked on IBM computers and we had a math library called BSLIB, Boeing Scientific Library. On the commercial side there was a mathematics group which I had found out about later. They worked on control data equipment and they had a library called the Library of Mathematical Routines, LMR. The two groups had very little communication with each other, even though both were a lot of Ph.D. mathematicians working on sets of problems. I worked on mathematical software a little bit right when I joined. But about June of '69, I had been there about five months, I got involved in a project where we were trying to build a mathematical model of the flow of electricity through large electronic systems. It was for a missile application. Someone came to me with a problem and said, "Can you help me solve this linear algebra problem?" which was perfect. I helped them immediately because I could apply some algorithms that I knew, but the more I looked at this problem the more I realized this had some characteristics that were kind of interesting. First of all there were lots of zeros in the matrix, and I remembered Bob Brayton in 1967, and I said, "Ah, this is why this is important," and I called him up. I said, "Do you have any software that I could get a hold of that I could use to help address this problem?"

[Tape 1, Side B]

Bob Brayton told me, "We do have some software that captures this, but we can't give it to you because there was just recently a ruling within IBM that no longer allowed us to give away our software."

HAIGH: Would this have been about 1970?

ERISMAN: It would have been late ’69 or early ’70. I don’t remember exactly, but it was very close to that. But Bob said to me, “However, I can still help you because Chris Pottle at Cornell University got a copy of the software before that. Why don’t you call him up and see if you can get it from him?” So I did and, in fact, I got a copy of that software program, the very first general purpose sparse matrix software. Somewhere in this house I still have the deck of cards that he sent me that embodies the Generate and Solve, GNSO, software for sparse matrix computation. We applied that to solving this problem, and we very quickly got to the point where we could solve two to three thousand complex equations simultaneously, which is really bigger than anyone had been solving.

On the basis of that, IBM contacted me and asked if I would give a paper at a conference they were doing on sparse matrix computation. So in 1972 I went back to IBM Research Labs and gave this paper on our application of sparse matrix computation. I think that was the first paper I published, and it really started my career in sparse matrix work.

This was published in a proceeding. ["Sparse Matrix Approach to the Frequency Domain Analysis of Linear, Passive Electrical Networks," *Sparse Matrices and Their Applications* (Donald Rose and Ralph Willoughby, Editors), Plenum Press, 1972, pp. 31-40.**]** and was my first invited talk and the first published paper.

After that we wrote one for the IEEE as well["Exploiting Problem Characteristics in the Sparse Matrix Approach to Frequency Domain Analysis," *IEEE Transactions on Circuit Theory,* May, 1972, pp 260-264, with Gary Spies].

HAIGH: So what kind of impact did the ability to solve these larger systems have on the work of the people within Boeing whom you were assisting?

ERISMAN: Well, what had happened, remember I told that from ’69 to ’71 Boeing went from 105,000 to 35,000 people. But we had a contract to understand this electronic system, and the algorithms that we developed were key to being able to do that, and our contract grew. So I am sure that one of the reasons that I as a rookie hung around the company during that terrible downturn was because the software was really key to that contract and one of the few contracts that was growing at Boeing during that time frame. So it had a huge impact there. It was much later that we developed this into any kind of general purpose software for a lot of others to use, and that did come later. But at first it was simply head down focusing on the characteristics of that problem looking at the software that we had acquired through Cornell, tailoring the two, and solving this problem.

HAIGH: So in your first job where you were doing this internal consulting work, what was your relationship with the computer center?

ERISMAN: The computer center, again, was something we used on a regular basis. Computing at Boeing in 1970 was really crude. You would take a job over. If your job was deemed important by somebody you could get it back the next day, if it was deemed unimportant it would be run as a fill in. I remember a colleague getting a job back a week later, finding he had an error in his Fortran statement and then resubmitting it, so you can imagine how horrible that would be from a time point of view. Later as we got going on this project we started getting higher priority where we would submit a job and guarantee getting it back in an hour and a half, which was a huge breakthrough for us.

Then about 1972 or so, we discovered a computer that had been surplused. It was an IBM 360 model 64. It was almost a minicomputer because it was nothing like the mainframes. No one knew exactly what to do with it, and we found out about it. So we actually went to the computer center, we got a pass to go in, and we sat there and we fed our own jobs in. I learned to watch the progress of the computation on this matrix calculation by watching the blinking lights—I could tell exactly what it was doing, what phase it was in the computation. It would spit out the printout. We could make changes in cards and recycle this thing in real time. As others found out about it you had to sign up, but we had an important project, so we would get this computer from five at night until eight in the morning and we were the only ones that used it. So we would work all night, I remember one week where we worked all night every night, we went home and slept three or four hours, and then went back. We were very hands-on with the computer then.

HAIGH: You have mentioned that at that point there were two libraries. Did that correspond to two physical computer centers?

ERISMAN: Yes, one was the Control Data center and one was the Univac that became the IBM center. Then in 1970 Boeing Computer Services was formed, and one of the steps was to take these two mathematics group and put them in to one organization. So there was a force fit between the two groups, and the person that headed the library of mathematical routines, the CDC side, the Boeing commercial side, was appointed the head of this group. His name was Sam Jacoby. So he put the mathematics organization together as one, and we started the project. A little later, I think it was around 1973 or maybe ’72 I was given the responsibility of bringing these two libraries into one. The difficulty was not so much in selecting the algorithms; it was more that at this time the mathematical software was embedded in applications everywhere, and the user interface to the library had to be stable. So we had to do a transition over a period of time to, first of all, choose the best algorithms, and then to keep in place the old calling sequences so that people could continue to use the software even though the underlying software had changed. It was a gradual process that we worked on over a period of years.

HAIGH: So I will ask you some more questions about that in a minute. Just before we move on to that could you describe what Boeing Computer Services was?

ERISMAN: I think it was originally conceived as a time sharing company. Computing was hot, Boeing was not, and the fathers at Boeing decided it would be great if we could diversify and start making money in another way. They formed Boeing Computer Services in 1970?

HAIGH: I think I have the date 1971 in your article in the Cowell volume. ["The Boeing Mathematical Software Library," *Sources and Development of Mathematical Software* (W. R. Cowell, Editor), Prentice Hall 1984, pp. 321-345, with K. W. Neves and I. R. Philips.]

ERISMAN: Okay, it could be. 1971 would probably be right. Even though I was working on this project for the missile system, I was on loan to that activity out of this mathematics home group. So I would spend some time back with my colleagues, and when Boeing Computer Services was formed this linkage took place.

HAIGH: So was the whole of your mathematical consulting group folded into the computer services?

ERISMAN: Yes.

HAIGH: And both of the existing computer centers?

ERISMAN: Yes, ultimately the Control Data center continued to exist and the IBM center. But over time the Control Data system, later the Cray system, became the focal point of scientific computing, and the IBM machines for data processing, although there was some scientific computing that continued on the IBM machines.

HAIGH: You had said that BCS was conceived as an entry into the time sharing market. You also said that you were physically running the jobs over to submit them to the computer center. So at the time BCS company was founded, did Boeing in fact have a time sharing capability?

ERISMAN: Through the IBM system they would basically sell time on these computers. I don't remember exactly when the terminal hook up into the system started happening, but it was some time in the 1970s because somewhere I have a picture where I was sitting at the kitchen table with a wire running out of this crude terminal. It looks not that different from your computer, but it had no smarts in it at all, it was a dumb terminal. I was doing some research and running some jobs on our computer remotely in the early '70s, so I know that capability existed early on. I think it was actually Boeing Computer Services Incorporated, and they incorporated in Dover, New Jersey and had their headquarters there. The idea from a Boeing point of view was we can make some money on this diversification and we could also manage our own computing environments, so we could get double leverage off of this.

HAIGH: So how successful was the organization in attracting external customers?

ERISMAN: They attracted external customers. They actually grew to be nationwide, even worldwide. I remember visiting the Boeing Computer Services offices in London one time when I was there. They did fine job at attracting customers. I don't think they ever did very well at making money.

HAIGH: Was the external customers exclusively for scientific and engineering applications, or was BCS also trying to sell data processing?

ERISMAN: No, data processing. Data processing I would say was the bigger of the applications.

HAIGH: Even for external customers?

ERISMAN: Yes. But Boeing, and computing, went through interesting and parallel tracks during that time. In the 1980s Boeing bought its first Cray, and then had a huge offering to the oil industry providing supercomputer calculations for oil companies all over the country. So that became a very heavy scientific offering. Also in the '80s we formed a software division, and they decided software was the key to the future and Boeing ought to be offering software. So one of the things they did was to look at all of the software work that was going on in Boeing and they said is there any of this that could be packaged and sold externally? We had two things that came out of the math group. One was a control systems analysis simulation piece of software called EASY5 which is still being sold to this day. The second one was our mathematical software library, BCSLIB, which at the time was integrated across the machine types. There is the product that was sold because PCs were coming along, so we actually started selling this library, probably never really in competition with IMSL and NAG, although we kind of thought that at the time.

HAIGH: Well I will ask you about those developments in the late 1970s and ’80s later on. So you said that in terms of external customers that Boeing Computer Services was reasonably successful in attracting them if not in making money. Now how did this spin-off affect your relationships with internal users? Did it create barriers?

ERISMAN: Sometimes it did, actually. Boeing was nominally a closed shop, and there were these large programming groups within Boeing that did the programming for the rest of Boeing, and they were assigned to Boeing Computer Services also. So in some ways it became an “us versus them” kind of thing. There were some funny ways the charges worked. For example, if someone in Boeing wanted to buy the services of someone in Boeing Computer Services, they had to pay their overheads. Whereas if they used someone from their own organization they didn’t pay any overhead. So it created this great disparity. But there was a policy directive that said you have to use these people. So it created some animosity. It was never very well done in my view. I think some of us, though, saw Boeing Computer Services as this is an exciting new venture as opposed to a staid old venture, so we were really excited about it.

HAIGH: Now you already mentioned the creation of the library, and I am about to ask you some more detailed questions on that. Just before we return to the details, I was wondering if the provision of this library was seen as a crucial part of the BCS service offering in terms of attracting customers.

ERISMAN: It was, because having efficient and accurate mathematical software meant that anyone who used our computers would have immediate access to this as well. So when someone developed an application on one of our computers, they could incorporate one of these mathematical library routines and therefore make their own application much more efficient and it would be much easier to write this. So it was used by both internal customers and external customers. A problem we had to deal with early on was if you are developing an application for someone like an Air Force contract, you have to deliver that application. What do you do about the mathematical software? So we had to go through this idea that we could provide releases in object code so that anyone who developed the software using a library routine would then be able to continue to operate the software and deliver the library routine as well.

HAIGH: Do you think the provision of that library set Boeing Computer Services apart from competitors such as Tymshare and Comshare?

ERISMAN: For data processing it probably didn’t make that much of a difference. I was primarily thinking about scientific computing. But certainly when we got our first Cray, and I think this was early 1980s, we immediately set about to tuning algorithms that would take advantage of the vector processing. When the oil industry started using our Cray, our big sales pitch to the oil industry was that you can get more effective use of a machine by using the mathematical software that has been tuned to the machine. Several of us from the mathematics staff spent months of our lives traveling to Houston and Tulsa, Oklahoma and various places giving talks on mathematical software and super computing. In that case we used it very specifically as a differentiator.

HAIGH: On the scientific and technical computing side, what organizations would you have seen as the main competitors to Boeing Computer Services?

ERISMAN: I guess it would be that a lot of people would use the computers at the Air Force Weapons Lab and at Livermore. They also made their systems available, I don't remember exactly what the business proposition was. Then later there were the supercomputer centers. The one at Minnesota was a really big one, they acquired the early Cray and early Cyber systems, so they had a variety of things that people could use. The University of Minnesota had a really big operation run by a fellow named John Sell.

HAIGH: So from your perspective then, the external users would be people who needed the capabilities of these large machines. You wouldn't be going more against general purpose time sharing like Tymshare and Comshare.

ERISMAN: Well, we certainly did with our IBM offering, but I wasn't as much involved in that. The scientific computing was the area that I was most interested in at the time.

HAIGH: So the scientific computing offerings were for these large problems, not just for a scientist or an engineer who needs some computer time and doesn't have access to a computer locally?

ERISMAN: Right. Well, we made a big pitch. I remember we were committed in the early days of supercomputing to be state of the art, and a couple of oil companies did most of their computing using our system because they said Boeing has committed to keeping up with the state of the art. They have the software, they have the people and the expertise to support us, and that is something we won't have to do if we use their system.

HAIGH: How would the time-sharing services have been sold? Was it by the hour in terms of computer time, or would you sell some kind of package?

ERISMAN: There was a complex algorithm that involved the bandwidth connection, it involved the computer time, it involved the memory allocation, it involved the tapes and disks—it was very complex and it was something that people always would game. You would figure out what the algorithm was and then you would figure out how to get your job done as cheaply as possible, exploiting the algorithm. The worst case of this was at the Air Force Weapons Lab where they charged only by the CPU hour, so we found by doing certain things with memory we could actually greatly reduce the CPU hours generated and tie up the whole machine, so ultimately they also had to change their algorithms.

HAIGH: So the pricing was on the basis of the use of these technical resources?

ERISMAN: Right.

HAIGH: Would they receive assistance and support bundled with that without additional charge?

ERISMAN: Sometimes, and sometimes they had consulting contracts where our people got involved and co-developed some software with some of the oil companies.

HAIGH: In practice, was it the case that most of the external customers were working on a time-sharing basis or would they deliver tapes with code on them and run these big jobs and have the results back on tape?

ERISMAN: They were working more on a time-sharing basis, I would say.

HAIGH: So I think that concludes the big picture for Boeing Computer Services in these early days.

ERISMAN: Maybe the one other thing related to mathematical software. In 1974 we did a paper at Purdue University on mathematical software, because by this time we had put the two libraries together, and in putting the two libraries together we had conceived of something we considered more than a library. We had the idea that you could go the way LMR had gone, which was a seven volume set of software which answered every possible question, but it was really difficult to find for a novice user what to use. NAG kind of went that way to with a kind of multi volume set of software. We said what we want is a single volume with a limited number of subroutines, and this would be our level one library, and this would be available on all the computers for anyone who wanted access to the simplest thing. Then we would start looking at more specialized software, and our mathematics staff was out developing particular applications for fitting surfaces and curves and doing optimization and all sorts of special function calculations. Not everyone needed those, but we wanted to reuse those. So we packaged these in a way that if people looked at our library and said, "Well, this is great, but you don't have a subroutine that does this," then we could say, "We probably do," and they would find it in the second level. Then we had a third level which was there was this public domain software that was being developed that some people just wanted access to, such as the DOE software. We would make that available and we would call that level three. So we were really offering a service. We had consulting that went with this, we tracked the usage, so that since the libraries were installed on mainframes we would know how many times each routine was accessed and we could put out statistics on the most commonly used software. We could identify the biggest user, we put out a newsletter, we would do seminars on mathematical software. So that was really the heyday of mathematical software kind of from the mid-'70s to early-'80s I would say.

HAIGH: So let's talk in a little bit more detail about the library. You mentioned that originally there were these two separate libraries, and with the formation of Boeing Computer Services the job was begun of merging them. How would you describe the quality of the routines contained in the libraries in 1971?

ERISMAN: I think to this day the quality is very high. We had some very good people, and they were very concerned about both accuracy and speed. I was always reminded of the quote from a British mathematician Joan Walsh, "If the answers are inaccurate, it's no consolation that the method was fast." [chuckles] So I think we did a good job of trying to look at efficiency of the algorithm and accuracy, putting out appropriate error messages, and figuring out how to make sure that a user wouldn't get lost and get strange results that weren't valid. I think the big job in the merger of the two was to select one way of doing things, to keep around the old calling sequence for people that had this imbedded in their software, and to create this transition.

HAIGH: You mentioned the evolution of the multi-level system. Back in 1971 prior to the consolidation of the two libraries, do you think all the code that was present in them at that point would have been developed internally within Boeing?

ERISMAN: It would either have been developed internally within Boeing or it would have been acquired from outside and then stabilized. By that I mean we would test it, we had some rigorous testing, we would make sure the error messages were right, we would make sure the calling sequence worked and were consistent with a template that we had created. Certainly we drew from LINPACK and EISPACK, the early linear algebra software, out of DOE.

HAIGH: Those would have become available a few years I think after the formation of Boeing Computer Services.

ERISMAN: Right. But as we went along, as the software became available, we would look at it and if they had something better than ours we would take the software, deal with the calling sequences, deal with the error corrections and all that kind of thing, rigorously test it, and then we would install that as well. Probably in the early days it was all internally developed, but later it was using the best we could find.

HAIGH: I think from what you said and what I saw in the article in Wayne Cowell's 1984 volume "Sources and Development of Mathematical Software," that the original platforms were the IBM and Univac computers that had been there before and a CDC machine.

ERISMAN: Right. The CDC was on the commercial side, the Univac going to IBM was on the military side, and when BCS was formed they made one computer center and then we made one library that worked across all the systems.

HAIGH: You mentioned that there would have been some public domain code that might have found its way into these libraries. Would the IBM SHARE user group have been the source of that?

ERISMAN: Perhaps. Yes, I would say that very likely. And published algorithms, in the early days certainly before the *ACM Transactions on Mathematical Software* there were books on numerical analysis that would contain algorithms. People would start there and they would say, "Okay, let's use this as a starting point, and then let's make sure that we can rigorously test it and that it works and puts out proper error messages."

HAIGH: And there was the Algorithms column in *Communications of the ACM*?

ERISMAN: Right.

HAIGH: Are you aware of any organized group for exchange of software on the Univac platform?

ERISMAN: The Univac was being phased out as I came in, so I don't know.

HAIGH: You mentioned already your experience with the sparse matrix code from IBM, and I will be returning to that topic later. I was wondering if you had had any other personal experience in these early days with software from outside Boeing.

ERISMAN: Well, certainly the LINPACK kind of software, but that was a little later. I don't remember the detail because I was very much involved with the one project over my first four years with the circuit simulation, but I would assume that much of the software was either homegrown or borrowed. The exchange in software at that time was really very different than it is today. To me one of the defining trends that gave rise to the mathematical software community was the move from a sharing and free exchange environment to the idea of selling software. That was a pretty radical idea in the late 1960s, early '70s, and it gave rise to a very different way of thinking about software.

HAIGH: Well, in some ways the more fundamental move is not just that the software is sold but that the software is really packaged and standardized and supported, which would be a precondition for commercialization. But similar developments in terms of the packaging and standardization of free software like LINPACK and EISPACK seem to have been happening in parallel with the commercial libraries. Both sides were advancing.

ERISMAN: Right. That was a way of getting some very fundamental software tools out there and it was great. But in parallel to LINPACK and EISPACK there was something called SPARSEPACK, which came out of Waterloo, probably at the end of the 1970s. Then the software out of Harwell initially was free, but then ultimately they decided to charge for it. Iain [Duff] can tell you exactly when that happened. Then there was something called PDEPack. I don't know if you have come across that in the course of your time, but Richard Sincovec and Neil Madsen developed software for solving partial differential equations and that was the first of that software that was packaged. That one was sold as opposed to given away, and I don't remember the exact timing. Richard Sincovec is that the person I went to graduate school with. He later worked at Boeing and he was instrumental in software for PDEs.

HAIGH: So all those developments were taking place in the 1970s. When I talk to people who were involved in the 1960s, for example in national labs, the striking thing to me is how little exchange of software went on. Mostly things were developed in house, and if they weren't developed in house they were really rewritten and tested in quite fundamental ways when they were borrowed. Why do you think it was that people in the 1960s hadn't made that push to package and standardize software, and seemed by and large happy creating things in house?

ERISMAN: I think a part of it is there is a certain thrill of creating something, and I think that forever continued to inhibit the use of mathematical software in some ways. It is kind of fun to do this, and I spoke on that at a number of meetings. When I taught numerical analysis in the mid-1970s at the University of Washington I would have my students write a program, but I would ask them to write a comment across the top of it that said I will never use this in an application. I said, "You are not good enough in a beginning course to write something that would really ultimately be useful." So you need to write it in order to understand how the algorithms work, but then you need to use someone else's software. So this was in the mid-1970s. Another point: this progression with mathematical software is much like the progression in computing in general. Early, people programmed in machine language. The first Cray computer had no operating system. But as time went on, the user became less tied to the particular machine through high level languages, then libraries, then applications. So we have gotten used to the idea of working at a higher level.

HAIGH: So I have here from 1979 from the book *Performance Evaluation of Numerical Software* edited by Lloyd Fosdick, published by IFIP, the summary of a panel discussion that you chaired called "The Use of Mathematical Software Outside the Mathematical Software Community: A Panel Discussion". You list a number of reasons there, including awareness on the part of scientists and engineers, understanding of the difference between something being right from a textbook and a quality piece of software. You also mention a lack of availability in software that in some cases the lab might have received a tape of EISPACK or IMSL and just not put it into the hands of the users.

ERISMAN: Right. A computer center would have to actually make something available, because at that time people were operating off the mainframes, so the resources you had were what the computer center decided to host on the computer. So the lack of availability was true on that platform. We looked in the 1970s at the possibility of stopping our library altogether and going with either NAG or IMSL and there were two deterrents to that. One was understanding this release process—if we embed their software in our application, what right do we have to use the software? We couldn't get that straight, and that's an availability question as well because the software is not really available to the application if you say, "You can use my library, but anyone who uses your application will have to buy my library in order to continue to use it." The second thing was a theme that I established early on in this conversation. As you start encountering very, very difficult detailed problems and you develop very specialized software, how do you make that available for reuse by other people? I remember some long discussions with Brian Ford at NAG on how would we make sure that some of the very complex curve and surface fitting software that we were developing that Boeing engineers use, how do we get that into your library so that people would be able to then use it? The answer is we couldn't, so we would need to have a library to share those kinds of things in any case, driven by our own applications. So that was one of the deterrents there. John Rice actually helped me put the panel together here. He had the contacts of the person from Russia and from various places. It was a fun panel discussion.

HAIGH: Two other areas you mentioned are usability of the software, and the appropriateness of the selection of subroutines for a library. I think both in terms of making the library powerful yet accessible to ordinary users.

ERISMAN: Right, and that is the kind of thing we were trying to juggle with our level one, level two.

HAIGH: About this level one concept of having a simplified consistent comprehensive version of the library. Do you know if anything like that had been attempted in any other libraries?

ERISMAN: I don't think so, and I don't know why; it seemed to me like it was a good idea. We talked about it in 1974 at the software conference at Purdue where I gave a paper. I remember Vel Kahan saying to me at that meeting, "You don't really have a library in the usual sense. What you have is a mathematical service that includes the software, but it includes the consulting and discussion, it includes the awareness." We even put out a newsletter telling people about new features, new ideas, how to use software, and so on. So the software was really only a part of the total service that we were offering there.

HAIGH: Were you aware of the IBM package SSP and its successes?

ERISMAN: Yes, I think we were. I think ultimately because we had our own software that people could use and release and be supported, it wasn't really relevant to us. I do remember something about that.

HAIGH: Well, SSP, which I think may have been released as early as 1966, was IBM's first supported package bundled with the 360 hardware, and I think there was some successor packages from IBM. I believe one was called SLMATH.

ERISMAN: Right. I remember it now when you mention it.

HAIGH: I know that the quality of some of those routines was viewed quite critically by some members of the mathematical software community, but I was imagining that they might have a similar kind of objective in terms of supplying a reasonably comprehensive but basic set of routines that would be accessible to all users.

ERISMAN: That could be. I think the decision that we made in going to these multiple levels was very consciously to not just offer the level one, the simplified stuff, but to make sure that people had a starting point from which to think about this and then go deeper into what they were looking for. I don't know that anyone else did that. Maybe they did. I don't claim to be an expert on that history.

HAIGH: Would all the code that made its way into the level one library have originated within Boeing?

ERISMAN: It would have been tested, standardized for error messages, documented by Boeing. Whether or not it was based on a framework that started somewhere else was not really the key thing.

HAIGH: So some of the code might have started somewhere else?

ERISMAN: Yes.

HAIGH: So can you maybe talk through the process by which a routine would enter the library. If there is any specific case you can remember then use that as an example, otherwise just describe the process in general terms.

ERISMAN: I will have to do it in general. We would identify an area of need, we would look for candidate software, and we would test the software. We developed over the period of time a very detailed test suite, and ultimately this test suite was automated, which meant that whenever an operating system was upgraded, whenever there were any kind of changes in the environment, we would run this whole test suite and look at error messages to determine if everything had gone well. It would go though a rigorous process of documentation of the error controls and all those things. We had within the test suite things that were designed to test the limits of our software, so we would have cases with various degrees of ill-conditioning, and we would look at the kind of errors that could be generated. The test suite was something that we worked very hard on, to make sure that not only was the software reliable when it went into the library, but also that it remained reliable over operating system changes.

HAIGH: Now you had implied that an addition to the library would be prompted by recognizing there was an area of need. Was that always the case or would it sometimes be that you came across a nice routine that had been produced within Boeing and decided to make it part of the library?

ERISMAN: Usually that would start with making it available through what we called this level two, so that it would be available, but it wouldn't be documented in the primary document that people got. They could find out about it through calling our help desk, and when something became strongly used then it would become a part of the main library.

HAIGH: I think you had suggested that users of the service were able to include compiled versions of the library code in applications.

ERISMAN: That is correct. And release those outside the company.

HAIGH: Now did that provision restrict your ability to incorporate externally produced code into the level one and level two of the library?

ERISMAN: That was a real need for us, and one of the barriers, as I mentioned, in using the NAG or IMSL is that we were developing these applications as a means to some other end: to satisfy a contract, to do some certification, to allow a partner to be able to use the software.

[Tape 2, Side A]

HAIGH: So you had just been dealing with this issue of the redistribution of externally produced code.

ERISMAN: Right, and because the application may be produced on a government contract, or may be part of a commercial contract where one of our partners needed to be able to use the software, we wanted to be able to make the software available to them, so that was one of the primary reasons we never switched. We looked quite a number of times at what we could do if we were to use a library other than our own.

HAIGH: So you mentioned that Boeing would be able to redistribute the object version of the routines. Would users of the library see the source code for the library routines?

ERISMAN: No, they wouldn't; we protected that. The reason for protecting it was not so much that we didn't want anyone to see it; it's that you can't possibly maintain software that other people change. If you actually give them the source code and they can fool with it, and they come back and they say, "Your routine doesn't work," well, what did they do with it? It was just something that we decided not to tackle.

HAIGH: Now one of the comments that I've read in some places is that mathematically sophisticated users would be reluctant to trust a routine if they weren't able to see the code. Did you find that was a problem in any cases?

ERISMAN: Sure, there was no question it would be a problem sometime, but remember what the library is for. It's not so our friends and colleagues who are mathematically sophisticated can examine the software and critique it. It's for people who don't really have the ability to write software to be able to do computations they couldn't otherwise do. So our primary audience was for engineers and scientists to be able to extend their own knowledge by using the software. Even there of course there are some people that say, "if I didn't do it myself it couldn't be very good." You always run into that. It is one of the deterrents to people using the software library at all.

HAIGH: Now one of the points you made to me earlier was that when you looked at the statistics for what people were actually using the library for, you found that things like linear interpolation, which wouldn't be on the radar of most of the mathematical software creators, were very widely used and that some of the advanced eigensystem routines were relatively little used.

ERISMAN: Right.

HAIGH: So maybe you can elaborate on that and how that kind of insight shaped the development of the library.

ERISMAN: Well, it does go to the point of the library in trying to raise the level of capability for a large number of people. Something like linear interpolation you wouldn't go too far wrong with I suppose. But by providing a routine to do this, you could do it very well and reliably and people wouldn't have to think about it. So yes, it certainly did shape it.

We also recognized, however, that you may get thousands of uses of that capability as compared to only a few uses of say an eigenvalue calculation. But the eigenvalue calculation would be much more difficult for a user to recreate themselves, so the benefit that comes from those few uses can still be high. We had to look at the complexity of the software problem and the uses and balance those two, recognizing that our goal is to create value for the company.

HAIGH: Do you think there was anything distinctive about the kinds of computing needs of Boeing's internal users that might have influenced the development of the library in one direction or another?

ERISMAN: There is something about an airplane in terms of its size, in terms of its life cycle, in terms of the number of parts. An airplane has three million parts; an automobile has three thousand parts. So just the pure scale. Then you get into the aerodynamic shape of the wing, which is very critical to its flight characteristics. The aerodynamic shape of the car is somewhat critical. So you get into the fact that you are really pushing the boundary in so many things, and I'm sure that that shaped our early need to solve large sparse systems of equations, it shaped our need to do very complex aerodynamic calculations, complex surface fitting. So I'm sure, yes, the applications are what shaped our work on the libraries.

HAIGH: So what did that mean in terms of areas of priority for the library in comparison say with Los Alamos with its interest in large-scale simulations?

ERISMAN: I was involved in the review of the Los Alamos computing division for a few years so I did see some of the computations that they did. They were complex, very difficult, and also different from the ones that Boeing did. Those differences make a difference, and I'm sure it shaped the software that the math people at Los Alamos put out compared to what we put out. I think the whole idea was keeping track and understanding what your users are trying to do and then trying to help them do it. That was the discussion I had both with IMSL and NAG when I had conversations with them, and I had many over the years. If we were to buy their library, how would we handle these problems? If we were to buy them as a company (and we actually pursued this a bit with IMSL), how would it be of advantage to IMSL's customers to be able to draw on the software development that was going on that was very applications driven within Boeing. Ultimately it never looked like a good deal for either of us and we didn't do it, but we did certainly have these discussions.

HAIGH: That would seem like a natural point to break for a while then.

Start of Session Two, conducted in the afternoon of July 6, 2005.

HAIGH: So continuing where we had got to, I think we had talked in reasonable detail about the creation of the library and the practices involved with it in the early days so we should now perhaps shift back to your own career. You had joined Boeing as an internal mathematical consultant. You had become involved particularly with this project involving large-scale sparse matrix methods, and you had explained that in 1971 with the formation of Boeing Computer Services your whole group had been folded into this new operation. Now according to your resume, in June 1973 you became Manager: Numerical Analysis, Boeing Computer Services, and you "managed a group of ten applied research mathematicians in numerical analysis." So how did that promotion come about?

ERISMAN: I think the person that preceded me in that position had left to take a university position, so they were looking for someone. The idea of leadership was of interest to me so I was interviewed along with someone else and ended up getting that position. The challenge that I took was to get a lot more people involved in things like SIAM and creating professionalism in the organization, encouraging people to publish. It was a really good experience. We started the seminar series during that time, and Michael Powell from England was the first speaker at our seminar series. Iain Duff came to visit the next year, or maybe it was later that year, and he was one of the speakers. John Reid came out to visit; he was one of the speakers. So we just kind of captured people as they came through Seattle, and then we had internal speakers as well.

HAIGH: Now you had mentioned that leadership was something that was of interest to you. Was that a feeling that you had had for some time?

ERISMAN: I had always felt that this idea that I have talked to you earlier about of establishing context for mathematics, of collegiality and sharing, of valuing people's ideas. All of those things were important to me. So having the opportunity to try and put them in practice was really a great opportunity and I enjoyed it.

HAIGH: So with the shift to supervising a group of ten other people, did that make a big change in the kind of work that you were doing on a day to day basis?

ERISMAN: It meant that I couldn't do the kind of project work that I was doing with day to day development of software and that kind of thing. But in many ways I felt that it was an opportunity to learn more broadly about numerical analysis, learning not just about my own area but the areas of other people who were working in the organization. I also felt that to be an effective leader I had to continue to publish and stay involved in my own area of the profession, so I carved out some time to write papers and I continue to write papers, primarily in sparse matrix areas, and give talks.

HAIGH: Did you continue to personally write program code?

ERISMAN: Not very often, although I had this one event corresponding to the last code I wrote at the Boeing Company. It was probably in about 1985, and I made an inquiry to see if they were still using the code I had written in the early 1970s. They said, “Oh yes, we use it all the time.” I said, “Well, the state of the art has advanced way beyond where we were then. We need to make some changes.” I ended up writing software to solve a particular problem, and I did it just in the evenings and I did it just for fun because I wanted to see if I could still do it. But, no, in terms of day to day work, I was not able to do that anymore.

HAIGH: Was there anything that you found challenging in that transition to a more managerial role?

ERISMAN: There is no question that when you have responsibility for other people, firstly personnel problems are tougher than technical problems; and secondly people are valuable. So it seems like there is more pressure, and in fact I did find throughout my management career that the idea of managing people was something that was rewarding, but also hard. I would come home at night and I would be really tired. I remember when I was working on the book with John Reid and Iain Duff, we were trying to do this book whenever we could. We would work on our own and we would get together whenever we could. They were going to both be in Boston and I flew to Boston where I was attending another conference, and I stayed over a weekend and we worked all day Saturday and all day Sunday on this book. I came away from those two days at work exhilarated instead of tired, and I said how can this be? It’s because I wasn’t dealing with some of the people issues which are tough.

I think I tended to take management positions for two reasons in addition to the one I just described to you, just to be a little more clear and honest. One reason I would take a position is because I would rather do it than work under someone else who might be another candidate, so I did it defensively. The second reason I did it is that each opportunity to take a new position gave an opportunity to learn something new. When I took on the numerical analysis group, I then expanded beyond numerical linear algebra to other areas of numerical analysis, and when I took on other positions I learned about other areas of technology which I enjoyed.

HAIGH: Now the group of ten numerical analysts, was that the same consulting group that you had been working in since you arrived at Boeing?

ERISMAN: Yes.

HAIGH: What was its relationship to the rest of Boeing Computing Services?

ERISMAN: At first, BCS was simply where we got our paychecks. We were not deeply involved in the things Boeing Computer Services did outside of the Boeing Company initially. That started to change in 1975. That is when I had the next promotion to managing a group of about 50 people. That grew rather extensively after I took it over. It was the applied math organization, and there in addition to the numerical analysts we took on some other areas of applied mathematics and statistics.

HAIGH: I will just ask a couple more questions on this period then before we move forward. So you alluded to the international seminar series that you organized. Were similar programs underway in different areas of Boeing, or was this something that was new for the company?

ERISMAN: I think it was new. I'm sure there are areas in aerodynamics which I didn't know very well until a little bit later that were doing this kind of thing. There were scientific people in different corners of the company. But the idea that we can learn from others is something that I've always believed strongly, and so the idea of having a seminar series where you can bring in the best ideas is just something that I felt really strongly about. There were not a lot of good models to illustrate that.

HAIGH: So one obvious benefit of that is that people would learn directly from techniques that the speaker was explaining.

ERISMAN: Right. But it's also true that when someone presents a seminar they learn a great deal as well. So it works both ways.

HAIGH: Were there cases where the seminars led to ongoing collaboration or a relationship developing between the speaker and work that was going on inside Boeing?

ERISMAN: Certainly that was true in my relationship with John Reid, and later with Iain Duff. John and I met at a conference in Cambridge, England in 1972. We developed both an ongoing collaboration but a friendship that continues to this time. Even our families got to know each other as we stayed at each other's homes on both sides of the Atlantic. It was also true for others as well. I think it was through a meeting with Gene Golub that I heard about a young man teaching at Johns Hopkins University, a former student of Gene's, and I went out to Baltimore and ended up hiring this person and he worked for Boeing for many, many years, John Lewis. So you get these connections of one person knows someone who knows someone, he has a student, and you build a community, and I think the idea of having the seminar series is in part about building community in addition to getting the ideas of the speakers.

HAIGH: Now the other thing it says on your resume for this period is that you managed the company-wide project in mathematical software? So was that the library effort? Was that part of the library?

ERISMAN: The library effort was managed by the numerical analysis group, so I became the de-facto program manager of that project and was so until sometime later when I hired another person to take that on.

HAIGH: So then your next promotion in 1976, according to your resume, was to become manager of mathematics and modeling within Boeing Computer Services.

ERISMAN: Right. The composite organization that had formed out of the math groups in commercial and defense was managed by Sam Jacoby, and I continued to work for Sam Jacoby in this promotion. He had the idea that we could create an organization that would not only provide these kind of mathematics modeling and computer science services to Boeing (and he had helped grow that area) but also could do it in other areas as well. Boeing at the time was interested in continuing to develop new directions for broadening itself beyond just an aerospace company. So we formed an organization called Energy Technology Applications, ETA, in late 1975, early '76. My boss gave me the responsibility for the mathematical modeling, including numerical analysis and statistics Another person was given a responsibility for data management, graphics, and some other areas of computer science. We went after the energy market. This is a time when Carter was President and they were looking at alternate sources of energy. We said the skills we have would bring value, so we started getting contracts with Department of Energy, with the Electric Power Research Institute, and really developing our outside business. This was the time when we kind of emerged as part of Boeing Computer Services developing business outside of Boeing. We continued to support Boeing, so then we had to walk the ladder between the two.

HAIGH: You mentioned that you had been involved in doing modeling with this ETA business. How does that translate into the promotion as "Manager of Mathematics and Modeling?"

ERISMAN: What happened was that in late 1975 my boss, Sam Jacoby, a colleague Jim Tocher, and I kind of created this new organization that would go after the energy business, and I was given a responsibility for much of what my boss had had, and he kind of grew another piece. So we created this new organization to go after the energy marked. My promotion came as kind of a byproduct of the new organization.

HAIGH: So as a result of the new organization the number of people involved in doing mathematics and statistics expanded?

ERISMAN: Well, it didn't expand right at that point because we had some other small elements of mathematics that Jim Tocher had managed and Sam Jacoby had directly managed, and I pulled the whole thing together, all of the mathematics pieces. Jim went off and created the new part in computer sciences and engineering and Sam also brought in some other resources to help us go.

HAIGH: So the initial result was a consolidation?

ERISMAN: A consolidation of other pieces that I had responsibility for. But then our business grew and the needs within Boeing grew so we expanded fairly rapidly. I think we had 40 people at the start; we probably grew to 80 to 100 people.

HAIGH: Your resume says "developed and managed a staff of 60 to 90 advanced mathematicians and statisticians."

ERISMAN: Right, so it grew to about 90 then.

HAIGH: So that expansion from 10 people in your group to 40 and then as high as 90, how did that change the way that you had to work as a manager?

ERISMAN: Well, it meant that I had some managers working for me for the first time, so I had an increase in leadership responsibility. I hired some really good people. I hired Richard Sincovec who had been with Lawrence Livermore, and later was with the Department of Energy, and is now at University of Nebraska. I also hired Ken Neves from Babcock and Wilcox in the east, and Jim Phillips from Washington State University. Sincovec in that time frame then started some recruiting at universities, and we brought in some really wonderful people with new Ph.D.s, many of whom are the core of the organization today. So it was really the start of hiring some very good people.

HAIGH: You were in that particular job from 1976 to 1984. So what do you remember the main challenges and opportunities facing your group were during that time period?

ERISMAN: Well, we were in a period of growth. We had very little internal R&D money, so we were basically a consulting organization that would scramble to keep itself funded. We also were then scrambling on the outside, so we were developing this business with the Department of Energy and with the Electric Power Research Institute, and I spent a lot of time on the airplanes traveling. During that period of time Boeing got its first Cray, and we were helping the Boeing company develop its Cray market because this is basically scientific computing and we had the credibility in that scientific computing market. So I remember a lot of travel, a lot of hours, and a pretty exciting time.

HAIGH: I will ask you in more detail about the Cray in a minute. Now, when your staff that grew to 90 people, was the library the responsibility of a small number of those people?

ERISMAN: Yes.

HAIGH: So there was a dedicated library team?

ERISMAN: There was a small number of people that were involved in things like help desk, documentation, software support, and standards for the library. But there were a large number of contributors, because whenever we saw someone who was doing mathematical software, the question would be how could we reuse this software. So how could we take something that someone had to develop for a particular application and make it broadly available to others? I can remember David Ferguson, his primary job was as a geometry guy and he was doing some really excellent work in representing complex surfaces for the airplane. Well this was geometry software that could be used by others, so we would take some of his time and fund David to move away from his project and package what he had so that it could be brought into this library and tested and properly documented so that others could use it.

HAIGH: Were the staff in this group all recruited with advanced degrees from universities, or would there be cases where someone would already be inside Boeing working an application area, and his or her interests would develop toward into mathematical software as a thing in itself?

ERISMAN: For some of the positions, which were primarily software and support, they were not necessarily Ph.D.s. But foralmost all of the research scientist posts, we were recruiting Ph.D.s. Sometimes they were from universities. I already mentioned Ken Neves from Babcock and Wilcox. He managed the software library project for a time. Dave Ferguson came from the

Aerospace Corporation, John Lewis came from a faculty position at Johns Hopkins. So we had this variety of sources, some people that were universities and other companies that would transfer in to work there.

HAIGH: Were there any cases where people who might have begun as developers of mathematical software applications in other parts of Boeing or as users of the library might have eventually moved into your group?

ERISMAN: There were a few cases like that. There was Ivor Philips who was a Ph.D. mathematician from the University of Washington, he was working in a programming group and he didn't like what he was doing and we found a match, and he came in. I think he managed the library until about two years ago. So that is an example of someone like that. There were also several disastrous cases. People in the Boeing company would hire someone, a Ph.D. mathematician, and then they would call me up. I got a number of these calls, "We have this person, he is brilliant, but he really doesn't fit where he is. You really ought to hire him because he is a Ph.D. mathematician." Some of those didn't out very well. One of them was a guy who, I don't think ever did any really good work and I ultimately ended up having to terminate him. It was a very difficult situation that I wished I hadn't been placed in that situation.

I won't mention any names here, but we brought in a person who had a Ph.D. from Oxford, who came in from another part of Boeing in Philadelphia. Wonderful reviews, and he came and joined us because he had all the right skills in the kinds of computers we were using, the kind of simulation, the kind of mathematical algorithms—all the right skills. He had been there about three months and his supervisor came to me and said, "This guy isn't working out." We ultimately found out that he had some mental problems. I went back and looked at his diploma he had given me when he came in, and I called John Reid in England and I just said John, what do you know about this college at this university? He said, "There is no college like that at this university." We did a little more research and I found out this was a mail order degree that he had acquired while he was in Philadelphia from a mail order house where he ordered his Ph.D., so we ended up firing him for falsifying his record. 15 years later Richard Sincovec was at University of Colorado and he called me up and he said, "What was the name of that guy?" and I shared his name, and he said, "I thought so. He has applied to be Dean of Engineering at University of Colorado, and he outlines his brilliant career at Boeing." So you meet real characters, and sometimes these transfers don't work out so well.

HAIGH: So beyond those personal issues, is there anything you can generalize about skills to be a good mathematical software package person versus being someone who might be good at developing an application?

ERISMAN: I think there is the discipline associated with documentation. Ken Neves used to have this wall hanging in his office and it had a picture of a small boy on a toilet, and it said underneath it, "The job is not completed until the paperwork is done," and this was certainly true of mathematical software. If you didn't document it for others to be able to use, if you didn't write it in such a way that it was intuitive, it was a problem. So I would say that software skill was a really critical one, and paperwork and documentation and a user orientation. We found some people that were very good in this area, and not everyone was.

HAIGH: Well, let's move back then and talk in some more detail about the expansion of the library to cover other platforms. So according to your article in the Cowell volume, a minicomputer version of the BCSLIB library was produced by 1978. Do you remember anything about the arrival of minicomputers on the scene?

ERISMAN: I do, and they created an interesting problem for us because more and more users were using the minicomputer, but the question is how did we get paid to install the library on the minicomputer? Because the main way we got paid for the main frame version was that they actually did levy a tax on their users, and so the library was one of these things that was taxed and we got the revenue to support the people providing the service and updating the library. So as you move to minicomputers, where does this tax come from? The answer is it couldn't come from the system because there was no system; it was a departmental thing. Ultimately what we got the company to agree to is that they would create an overhead category, which is where these minicomputers ended up landing. That allowed us to distribute BCSLIB to the minicomputer. But it was a very difficult transition. Secondly, we couldn't get all the usage data that we got before, so we couldn't manage it quite as well that way. We were never quite as sure as we were before as to who the users were, where they were, and how things were used. This was just made more difficult at the next stage of this same transition when we went to PCs because there you are involved in delivering a disk and what people were doing with it, and how they were doing it, was something that we couldn't measure.

HAIGH: Do you remember any technical challenges with making these routines run on small machines?

ERISMAN: We had spent a fair amount of time as we brought our libraries together from the Univac/IBM route and from the CDC route of saying we really have to think about this. I think the word I learned from Brian Ford in those days was not "portability" which is sort of a myth, but "transportability"—how do you create software that will run well on many machines with minimal amount of work, as opposed to run average on any machine, which was not something that anyone wanted. We had done a lot of isolating machine constants and understanding word length implications and all those kinds of things. So I think we were well prepared for that. I think it was really much more the issue of the administration of it than the technical challenge of putting it out there. The PCs became different because there we couldn't count on the Fortran compilers, and all of a sudden we had to think in terms of other languages for our library, and we certainly went through that transition of trying to go beyond Fortran to other languages, like C and C++. That was a much harder problem than the transition to the minicomputer.

HAIGH: I know that there were a series of language standards for Fortran, for Fortran 66, 77, 90 and so on. Did those create any challenges or opportunities for the library?

ERISMAN: Our goal was to first of all be involved in those, so Ivor Phillips, who I mentioned earlier, became a member of that Fortran committee and served actively on that committee, so Boeing was represented there. We tried to anticipate the changes in the standards. We tried to do things that were as vanilla as possible. We had had a bad problem in Boeing Computer Services in that the company had started making changes to the CDC operating system. So we had our own version of the operating system and every time the vendor then updated the operating system we had to do all sorts of updates to keep up, and we recognized that it just isn't worth it.

So we tried to establish some disciplines to anticipate transportability and anticipate future changes in the language.

HAIGH: It's clear from other interviews and from a number of articles that were published at this time portability became a major focus of the mathematical software community in the mid-'70s. Particularly the technical details of portability in terms of isolating machine dependent constants, of trying to create automatic tools to compile platform specific versions from a single code base, the use of a portable subset of Fortran and so on. I was wondering if there anything you think is specific or distinctive about the techniques Boeing came up with to address these issues?

ERISMAN: Perhaps when the Cray came along, we recognized that the vector constructs were such that they made a huge difference in performance, depending on how you implemented the algorithm. So we asked the question how are we going to maintain our machine independence and at the same time take advantage of the architecture? So in addition to isolating machine constants, we tried to isolate constructs. So we developed something which we called BCS/VectorPack, which was a collection of vector constructs that we could tailor precisely to the Cray and wring every ounce of performance out of. But there would be a transportable Fortran version for use on other computers. So by using these constructs we could then actually exploit the characteristics of the computer and, in fact, we ended up doing a fair amount of those in the early days in assembly language on the Cray computer. We did that in order to get the performance out because the compilers were not sophisticated enough, but we tried to isolate that from the rest of the code so that it would be very easy to kind of keep it in sync. We worked hard on that strategy.

HAIGH: So would that strategy be comparable to the BLAS system?

ERISMAN: Yes, in fact it was about the same time as the BLAS were coming along. I'm sure at the time I knew what the compatibilities were and what the distinctions are; I just can't tell you at this point. I think the BLAS defined the mathematical construct, the mathematical operation that you were doing, and then what we did is how we implemented those in the vector computer.

HAIGH: As I understand it, for the level one BLAS, the idea was the same that isolate the architecturally specific features and then just have a standard call in the application code, so that you can implement an optimal package for each machine.

ERISMAN: Right.

HAIGH: The level one BLAS which would have come along in the mid-'70s just focused on the inner loop, which was good for non-vector machines. But to fully exploit the vector machines you need to do the whole matrix. I think vector-matrix and matrix-matrix operations were level two and three respectively of the BLAS, and I believe those only came along in the '80s.

ERISMAN: Right. And we were developing something like that recognizing the constructs on the Cray. Then exactly how we merged what we were doing and what was happening in the broader community in the BLAS I don't know, but I know that we made an effort to do that. But we went beyond the BLAS as well, looking for any construct that made a difference computationally.

HAIGH: So you think this would have begun as a completely separate effort from the BLAS and eventually would have been merged?

ERISMAN: Eventually merged, or at least made transportable between what we were doing and what they were doing.

HAIGH: On the subject of portability, as level two and you have also mentioned level three of the library would include externally produced software. So did you worry about porting that, or did you just offer that only when it had already been ported?

ERISMAN: Level two was more specialized software that we had developed that would have clouded the ease of use of the basic library, so we worried about it there. Level three, though, was for convenience. We would make some general purpose stuff available to people and we didn't worry about that. It was a service so you actually could get access to it: "Here it is. User beware."

HAIGH: Back on the topic of the Cray, you had mentioned that this early commitment to producing a vectorized version of the library led to some new opportunities. Now, I saw in the Cowell volume a reference that the first Cray version of the library was produced in 1981 as a direct conversion of the CDC version. So can you outline what happened after that and how the business side developed?

[Tape 2, Side B]

ERISMAN: I think the first step when you get a new computer is to get it functional. So we converted the library the library in a rather straightforward way, based primarily on the CDC version. The IBM version had the single and double precision stuff and integer star 16 and all theses kinds of various word lengths. CDC didn't have any of those things, the Cray didn't have any of those things, and the CDC was a more scientific computer anyway. So we could get something going pretty quickly because we had been working on the transportability part. Then the next step was to start tailoring this. That is where we started isolating these constructs. We recognized how certain things would work very, very well in a vector mode, but you had to kind of artificially capture that vector mode. In sparse computation, for example, what later became the sparse BLAS, how you do certain things that are not packed vectors in the usual sense? So what we looked at that from an algorithmic point of view to understand what it would take to make this work really well on the Cray computer, and then develop these things which we came to call BCS/VectorPack, which was this collection of very high performance constructs that would interface with our library software but be transparent to the user. So the user would see the same interface, but it would get this performance.

HAIGH: So that was the technical level. You mentioned the exploitation of the power of the Cray created some new business opportunities, particularly with the oil industry. Was this sold as a new product with some kind of new brand name attached to it? Was there a marketing push specifically around the Cray?

ERISMAN: Yes there was, there was a huge push. In fact, Jim Phillips who was running the numerical analysis group at that time, or one of them—I think we had two of them and Jim was running one of them. He took on this job of making this thing with the oil industry work. We ended up having the spring colloquium for the geosciences that he organized, and we brought in speakers, not just from within our own organization but other speakers as well, and created seminars for oil industry people in Houston and Tulsa and Ponka City to make people aware of what the Cray could do for them and to make them aware of our services in particular. We tried to distinguish ourselves. In retrospect, we didn't do a very good job of this with respect to Cray itself because we ended up selling a lot of Cray machines to oil companies, and there was nothing really in that for us. But we did end up also building some long-term time sales partners that lasted for a number of years. In fact, in our parking lot at the Boeing Computer Services you can still see the remnants of the satellite dish that was there. Amoco in Tulsa was doing a very intensive amount of computing on our computer and they had a satellite linkup for transmitting data between their data center and ours. So it was pretty successful, I would say.

HAIGH: So what would you say the balance of internal to external customers was on the Cray?

ERISMAN: We had a period of time where the balance was probably 50/50, and then there was a period of time when the Boeing use on the Cray started to dry up. We tried to figure out what was happening and one of the conclusions we came to (and this was probably in the late 1980s) was that the accounting on our Cray computer meant that if a user wanted to use the company owned Cray they would have to expend money from their department, whereas if they used there own workstations (and there were pretty powerful workstations developing by that time), it would be free. So they would set up these jobs that would run over a weekend and overnight on their workstation computer to save money for their department, even though Boeing owned both assets so it didn't really help anything.

So I will never forget a meeting I had with a chief technology officer of the Boeing company at the time, Bert Welliver, and the president of BCS, Art Hitsman. I suggested the accounting algorithms were making our Cray system unusable, and suggested we charge every user as if the machine were full, so they would get a nominal charge and they would be sure of it; costs for the part of the machine that doesn't get used would be put in an overhead bucket. So this way if a person was the only user that week he or she wouldn't have to pay for the whole system. It assured people that they would get a reliable bill—it wouldn't be free like their department machine, but it would at least get it closer. Then I had a bet with them as to how long it would take to fill the machine. They said it would take a year and I said it would take two months. I was wrong: it took one month. The users filled the machine once they recognized that there was a fair accounting algorithm. So this is an example of how an accounting system can drive behavior in how people use resources in scientific computing.

HAIGH: Did you have just one Cray machine or was there a succession of them?

ERISMAN: We had a succession of them. We had a Cray XMP, we had Cray YMP, and another one later as well.

HAIGH: Now, you mentioned the marketing side of the push to the oil industry. Was there anything you had to do in terms of offering applications or strengthening any areas of the library in order to attract that business?

ERISMAN: We tried to use the library as a means for helping people get really efficient software that they would develop, but we had to go further than that, because by that time packaged software was making inroads. So we went to some companies in the oil industry that had software that other oil industry people used and we helped them host that software on our Cray so that we could offer that as a service for our customers.

HAIGH: So it was getting more into what they would call now as being an application service provider?

ERISMAN: Exactly.

HAIGH: While we are on the subject, is there anything else you have to say about the Cray and the oil industry? I think now would be the time to say it.

ERISMAN: It was a fascinating time during the growth period of the oil industry and the entry of the Cray. There was a tremendous demand for oil, there was a tremendous boom. I mean some of these people were wealthy beyond belief. Then we saw the drop in the price of oil and the horrible situations in Houston where it became really tough, and we kind of watched that happen right in front of our eyes. I remember the story that started circulating, ”What is the difference between a pigeon and a Texas oilman? The difference is that the pigeon can still make a deposit on a Mercedes.” [chuckles] So it was an interesting time to watch that field. We also got involved a bit with the pharmaceutical industry with our Cray, and I made a lot of trips to visit various pharmaceutical industries to try and understand the nature of their computation and what it is they could do, why they would want to use this resource. In doing this, it really gave us on our staff not only a view of the whole of Boeing, but a view of the whole of the scientific industry. We built partnerships with other applications providers like ANSYS and Nastran also. It was a great opportunity in scientific computing during that time.

HAIGH: Now you also mentioned that you finished up doing some work under contract for Cray, or licensing some things to them?

ERISMAN: Right. We licensed some software. We started with BCS/VectorPackand then we built some very rapid high performance linear equation solvers that Cray offered and we got some revenue from, and we also then embedded these under contract with various application solution people such as McNeil Schwindler and Nastran.

HAIGH: Did that become a significant source of revenue?

ERISMAN: It was significant to us. It wasn’t significant by Boeing’s standards, but when you are a an organization 250-300 people within an organization of 100-plus thousand people then what is significant to you doesn’t show up on the bottom line of the large company, but it keeps you going and it allows you to continue to leverage and do really good work, which is what we were hoping to do.

HAIGH: So during this period, who was your boss?

ERISMAN: Sam Jacoby was my boss until 1984. Then in 1984 this ETA organization was considered a vital resource in Boeing, Sam Jacoby went off to do something in the manufacturing field, and I was promoted to run the whole ETA organization, which was maybe

180 people. From 1984 to '90 I ran this organization, and I think I reported to seven different bosses during that period.

HAIGH: We will talk about that side of it tomorrow. So my last question just would be: from your descriptions of experimenting with the different kinds of service offerings and licensing software and serving the internal and the external uses, this sounds like a fairly opportunistic selection of activities. So what did you have to accomplish in order to keep your group intact and your managers happy and make sure resources would continue to come?

ERISMAN: We always said that there were two measures of our accomplishment. One was the measure of customer satisfaction which would show itself in continued work and continued revenue, whether from the internal sources that we had or from the external sources that we had. But the second measure was technical excellence. You can sometimes satisfy an internal customer with something because they don't know how good they could get it. So it wasn't enough to satisfy them; we wanted to look for a measure of satisfying some technical criteria. That was one of the reasons we continued to push people to publish. We had some technical reviews where we brought in outside reviewers to look at what we were doing, and our goal was to kind of keep a balance and make sure that we were not only doing things that people wanted, but things that were excellent. That was, believe me, a challenge. There were people in the company that said, "Why do you want to do these technical reviews? Satisfying the customer is what you need to do," but no, it's got to be more than that.

HAIGH: So because you had this captive internal user base, you also needed to keep publishing and to do these things for external customers to show that you were world class and could compete in the broader market?

ERISMAN: But I wanted the ability to also tell the Boeing market that we were world class, and if I couldn't say that I couldn't continue to look for the internal funding I was trying to get to keep the group healthy.

HAIGH: So the external things were important to win the legitimacy to ensure the continued internal funding.

ERISMAN: Exactly.

HAIGH: Well, that seems like a good point to break for today then. End of session two.

Session Three, taking place on the morning of July 7, 2005, taking place in Dr. Erisman's home in Belleview, Washington.

HAIGH: So picking up where we left off yesterday, it seems we've dealt quite well with the evolution of Boeing Computer Services, computer platforms, the arrival of the Cray, the continued development of the library, and the external service offerings that were made by the group. I wonder if at this point it might be appropriate to talk a little bit about the continued contribution of Boeing Computer Services and scientific computing to Boeing's own businesses.

ERISMAN: Maybe I could add one PS on the evolution of computing. Not long after the Cray came out and the Cyber 205 and that whole world, there came copycat systems, Convex and SCS were notable there, where they tried to do a mini supercomputer. They took the supercomputer model and created an environment that would allow a departmental level supercomputer. Boeing got involved in that. We had a lot of conversation with Convex about the idea of mathematical software with them. Ultimately I don't remember what finally happened. We ended up striking a deal with a company called SCS, which made a Cray mimicking mini supercomputer that sold for under a million dollars. The company looked like it had a really good opportunity, and it blazed out for a couple of reasons. One was the marketing. I think they hired all marketing people that had been selling mainframes and you sell mainframes to a computer center and you sell minicomputers to a department, so the marketing approach didn't work very well. Then there was the micro revolution, which continued on which attacked the problem from below with more and more powerful small processors. But we did put our library on the SCS; I believe we had it on the Convex as well. We had it on a Floating Point Systems machine, and we were adapting it to those architectures and looking for efficiencies using the approach that we described in the paper on computational kernels. So that was another piece of the evolution.

HAIGH: Let me just then insert the citation for that paper. So this is "Computational Kernels", AM Erisman, published in *Computer Physics Communications* 37 in 1985, pages 149-157.

So were you undertaking those porting efforts because you thought that you might want to use those computers yourself or because you were being contracted by the manufacturers to provide these capabilities?

ERISMAN: Both. We wanted the library to look as common as possible from an interface point of view for all of our users, so that whatever machine they were on they could get good efficiency and rely on the mathematical software. We had contracts with Floating Point Systems, with SCS, and I believe with Convex as well to deliver software that would run effectively on those architectures.

HAIGH: You are describing this work here as "computational kernels." Is that the same project that you referred to yesterday as BCS/VectorPack?

ERISMAN: It is kind of related to that, yes. It started with an idea of BCS/VectorPack that was kind of its first instantiation, but we went beyond that to computational constructs that really could make a different in the architecture and that would make a difference for mathematical software.

HAIGH: So it was a generalization of the same concept?

ERISMAN: Yes. In the meantime, there was an interesting process that was going on within Boeing. We had done the simulation we had talked about earlier for Minuteman, we had done the simulation of these large electronic circuits, we could get some test data to validate our models. But there was a theme that I think is still probably spoken of at Boeing today that "when you use a computational model, no one believes the results except for the person who built the model." There are a lot of people who feel that having a physical test is much better. And the statement went on, "But with a physical test, everyone believes the test except for the person who performs the test," because they know all of the problems associated with, for example, wind tunnels and any other testing environment where conditions aren't exactly true to where the final product would be. So I think the idea of gaining credibility in models has been and continues to be a really important area. But we started pushing this well beyond where we were. We started doing modeling of manufacturing processes, including the riveting process on the airplane. We were looking at how do we know what a good rivet will look like, whether or not there is too much force being used that creates cracking and we built models and we built tests, and we started getting understanding and all of this was being made possible by the tools in mathematical software, the power of the computers, and the insight of some of the scientists that we had.

The Boeing Company itself was starting to go through a process. It used to be the wind tunnel was the ultimate tool. This was the design tool that Boeing had. As you move forward the wind tunnel became the validation tool. In other words you do all the experiments and explore different regions of design on the computational model, but you do a wind tunnel test at the end. So you took the wind tunnel out of the inner loop at least and did all of the searching for new designs and new design space on the computer. That was an evolution that probably took through the 1980s and well into the '90s in terms of people gaining confidence and having a much more powerful role for computing. It's that confidence that is really the key to being able to use the models well as a part of the way you design. In spite of the fact that cycles are very cheap, there are still some people who say, "You have got to limit the number of computer runs you do because computer runs are expensive," instead of recognizing that you ought to do thousands of computer runs to explore things that you couldn't otherwise explore. There are some breakthroughs that have been happening in the design work where truly the computer is used as the exploratory tool, and that's based on the software, the power of the computers, and the scientists that are at work. So this evolution has really gone a long way from people being very skeptical about the results to this being a central part of the design processes.

HAIGH: So as the computer became more central to and more tightly integrated with the design process, did that change anything about the relationship between your group and the design engineers?

ERISMAN: We ended up working a lot more closely together. A lot of our people had very, very close assignments. Well, something happened in 1990 which I guess we need to get to; maybe I should mention that first.

In 1990 happened something happened with me personally simultaneously with something happening within the corporation organizationally. Boeing decided in late 1989 to select a group of technical fellows of the Boeing Company, and they wanted people to be selected that had a reputation inside and outside the company. They ended up choosing 12 scientists across the company, and I was one of those. That kind of changed a lot of perspectives, because it isn't that you are doing anything differently, it's that all of a sudden people say maybe we ought to listen to what this person has to say. And it did make a big difference. It was also around 1990 that the President of Boeing Computer Services said that what we need to do is form a technology research organization within the computing world. I was operating this ETA organization from 1984 to '90, after taking over for Sam Jacoby. I was asked if I would take that organization, another organization that was working in some areas of R&D within another area of Boeing Computer Services, and a third organization called an advanced technology division which was focused at artificial intelligence problems. Out of these three organizations I created one technology organization that I directed. That happened in 1990.

With that, we for the first time got company funds to invest in R&D. So we were no longer just scrambling for customers to fund everything that we did, but we were funded fairly well. I would say about 50% of the organization was covered with R&D funds, and that kept us out working with customers and government contracts with the other 50%. But that funding really did change the nature of the way we work with the company. So that was a critical factor. The library then became a part of what was called the Mathematics and Computer Technology Organization, which was a part of Boeing Computer Services, at least until the point when Boeing Computer Services ended as a separate division.

HAIGH: So as you brought that up then, let's go back over a couple of those points in more detail. Looking at your resume from 1984 to 1989 you were Director of Engineering and Scientific Services. Now you are calling that ETA. Was that the informal name for it, or maybe the old name?

ERISMAN: 1984 to '90 was a very turbulent time for this organization. In that period of six years I think I had seven or maybe eight bosses reporting in different parts of the organization. No one could quite figure out what to do with us. And my focus during those six years, if I look at any accomplishments maybe there were a few others, but the main one was keeping the organization alive and keeping mathematics alive at Boeing through this scurry of reporting to different bosses. So as a result, the organization used to be called Energy Technology Applications, ETA, then it was called Engineering Technology Applications, ETA, then Engineering and Scientific Services, and Scientific Computing, and I think even several other things. The name changed as the management of the organization that we moved into changed, and it was an extremely turbulent time.

HAIGH: Then things picked up in 1990?

ERISMAN: Well, in 1990 in part because of the person who decided that we needed a technology organization within Boeing Computer Services, that was the President of Boeing Computer Services, his name was Art Hitsman. He was a very interesting man, he was an engineer by training, he was very rough, he was very gruff. It was always "show me", but he was one of the smartest people I ever met and I gained tremendous respect for him and he gained tremendous respect for what our organization did. It was a wonderful period in the growth of our organization because Art Hitsman saw the kind of things that we were doing, became an advocate for them. Another person, the Corporate Vice President of Research, Bert Welliver, was an advocate with Art. Together we got funding, we got recognition, and we established the organization and it was a life changing kind of time, after that six years of great turbulence. Both of them were great people.

In fact, one time during that period, 1990 to '96 or so while Art Hitsman was the President, someone elsewhere in the company came to see me and they said, "Art Hitsman won't let us go forward with our project until we talk to your organization." Another time someone said, "Your salesman in the President's office wants us to talk with you about this." So he was a real advocate and it was a real positive time with the organization.

HAIGH: Now the title that you have on resume for that period is Director of Technology.

ERISMAN: Yes. It was called the Technology Organization within Boeing Computer Services. I don't remember the exact time frames here so I'm going to guess, but Boeing Computer Services kind of recognized its time was past, it was no longer selling outside except in large government contracts, and ultimately they spun that piece off to the government military sector of the Boeing company. So we became a part of the internal computer services organization that Art Hitsman was President of. But the organization was simply called Technology and I was Director of Technology.

HAIGH: So why did Boeing Computer Services business gradually fade away?

ERISMAN: Time sales were over; that was one thing. Boeing Computer Services had a president that came in from IBM named Bob Dryden and he came in in the early '80s and he said "software is the thing." He is the one who formed the software division and that's how we started selling BCSLIB among many other products. We didn't execute on the software division. It was just pure and simple: the market was there as had been predicted, the execution was not there, so throughout that whole thing we were spending a lot of money and not making a lot of money. We had products in the oil industry, we had PC products, we had all sorts of things that were bad ideas. As a result, in the very early '90s Boeing Computer Services stopped being in the commercial business because we weren't selling software and we weren't doing time sales. The Cray sales had peaked at that point and people got their own Crays, so it no longer really made sense to continue.

HAIGH: Well, let's just talk a little bit about efforts to sell the library itself as a product then. You have a copy of it, one of those made in 1980s, strong cardboard boxes that PC software used to come in.

ERISMAN: Right. When the software division was formed I remember going to the person who headed that division and telling them that we had a couple of pieces of software that we were selling a little bit outside: BCSLIB was one, and a simulation package based on some mathematical software called EASY5 was another. I remember this guy saying, ""We don't want your cats and dogs. We are doing great things here with the software division. We are going to develop a three-dimensional version of a spreadsheet." Boeing did that, it was called Boeing CALC. It was widely advertised.

HAIGH: Was that sold?

ERISMAN: Yes, though I don't know how many copies were sold. But ultimately he ended up our software and adding it to his portfolio. And when the software division closed its door the last two things they were selling were BCSLIB and EASY5. So it was kind of ironic, but these were solid products. Easy 5 continued to be sold until I think about two years ago, it was after I retired. Easy 5 was spun off to MSC Nastran and became one of their products. Whether the math software for supercomputers and for applications is still being sold I don't know; it was to the point that I left the Boeing Company.

HAIGH: So was the PC version of the library the only one to be sold as a product?

ERISMAN: Probably not the only one marketed, but maybe the only one to be sold.

HAIGH: For example, IMSL had demonstrated in the early 1970s that there was a market for mainframe libraries to be sold as products rather than as services. But it was only in the PC area that you made attempts to solve the Boeing library.

ERISMAN: We made some attempts to, but we never really had kind of the sales structure to sell the library. Our claim was that BCSLIB was rooted in applications, and as a result it would have more relevant capability, and that's in fact what we argued in a 1974 paper that I talked about much earlier, that we could take new ideas and turn them into software and then distribute them as a part of the library. But we were never really very successful commercially, though it was a big success for Boeing. We didn't have what it took to do the production, the support, the marketing, the sales; we just never had it.

HAIGH: So saying by it was more focused on applications, are you implying that it was something that could be sold more easily to end users than computer center managers?

ERISMAN: Or more attractive to end users, which might make a better sale to computer center managers. That was our argument, but we never proved it, and maybe it wasn't true.

HAIGH: So do you know what year the shrink-wrapped PC version was launched in?

ERISMAN: I don't. It was undoubtedly in the early '80s, so I would guess '82, '83, '84, somewhere in there. I don't know if there is a date on this box.

HAIGH: Just purely from the box, I would guess that that box is maybe 1986 or '87 because that was when PC software vendors settled down on those handsome sturdy boxes. I think Lotus 123 started it and that came out in 1984 and then in the years just after that its packaging style became a standard. Although IBM technical documentation also shipped in boxes like that just for the PC.

ERISMAN: Let me just look and see if by chance there is a date on this.

HAIGH: You opened the shrink-wrap now so the resale value would have plummeted. A good collector's edition.

ERISMAN: My collector's item I just destroyed, right! 1987. It says copyright notice printed in the US 1987.

HAIGH: Does it have a version number? Do you know if that was the first edition?

ERISMAN: Version 10 it says.

HAIGH: I bet that is for the library in general.

ERISMAN: Could be. I just don't know.

HAIGH: You said earlier that when it was offered as a service the users never got to see the source code. Now when it was sold as a product was there technology available to continue to do that, or did you have to distribute the source code versions to be compiled in with the programs?

ERISMAN: My guess is that we had the source code in here. I don't know exactly what the control mechanism was, so I don't know the answer to that.

HAIGH: It remained a library, meaning that you would have to write a Fortran program?

ERISMAN: That's right. I don't remember when we started doing the C++ adaptations, but it did remain a library, and that was both its strength and its weakness.

HAIGH: In contrast with MATLAB, which would have been gaining strength in the market during this period.

ERISMAN: Right. And EASY5, which was a simulation tool. EASY5 added a MATLAB like feature, so it had some of the linear algebra software from BCSLIB and that became a part of EASY5. EASY5 was a competitor with MATLAB and other things—we used to meet each other in the marketplace there.

HAIGH: Would EASY5 have been more specialized than MATLAB?

ERISMAN: Probably. It was much more deeply rooted in dynamic simulation, so it was built around people putting together a digital simulation of a circuit or a control system, that was its roots. Whereas MATLAB was rooted in matrix computation and added the ability to do the simulation, so they kind of grew towards each other from either side.

HAIGH: What individuals would be most strongly associated with EASY5?

ERISMAN: The founder of this was a fellow named John Burroughs. John was a control engineer who built a simulation for a customer way back in 1969 and had a vision for a product, and so he built a framework from which he could do other simulations. When Cleve Moler and I would run into each other at a meeting and talk about this, he would tell me how much better his product was then mine. I just kind of laughed. He had a staff of 150 people or some such thing, and we had I think five people that were doing EASY5, maybe nine at the end. They were very different pieces of software. So EASY5 was a different instantiation than BCSLIB, but they had relationships with each other, common software and that kind of thing.

HAIGH: I think you had said that you considered the products themselves were strong but the execution was lacking. How would they have been distributed?

ERISMAN: Well, ultimately the PC version was distributed in this box.

HAIGH: That's physically. What were the sales and marketing channels?

ERISMAN: When we worked with Cray and with other vendors like SCS and Floating Point Systems, we delivered the software to them and then they delivered the software to their customers. So Cray actually did the distribution of the software with their machines. So we really never got involved in the large-scale distribution except for this attempt at a PC version with BCSLIB. With EASY5 we had a couple of internal sales people and later a couple of distributors. Ultimately, I believe it is tough for a large aerospace company to take seriously anyone selling something for less than $1 million. Further, software sales people most places had reward systems and a culture that Boeing could not sustain.

HAIGH: So let's turn back then briefly to the impact of these technologies within Boeing. You mentioned that the computer simulation was becoming more and more central to the design process. Now was there a separate group involved in producing or adapting CAD tools for the basic design work?

ERISMAN: Right. Boeing made a decision that they would buy a CAD tool; they made a run at actually building a CAD tool. They invested a lot of money in it, that was another part of the software division fiasco. It was called AXXYZ playing on the x, y, z axis and coordinate geometry. Boeing spent a lot of money on it, and ultimately. AXXYZ was ultimately spun off to EDS in the late 1980s where it ultimately died, I believe. But before it had spun out, Boeing decided to go with CATIA while BCS continued to try to sell it commercially. I was asked by one of our Vice Presidents to consider taking this product on in my organization, and I said the best that could be done for it would be to keep some of the very bright people involved and to spin off the software. He didn't like that answer, so he kept trying to sell it. I think two Vice Presidents lost their jobs trying to make this go.

The way our mathematics staff got involved was through the geometry software in our library. It was used to smooth surfaces, to interface between CAD systems, and analysis systems. So we added a fair amount of mathematical software and some of the big process, but the design software itself was CATIA. As I got involved with this technology organization in 1990, we had some other people who were doing some really interesting work in computer graphics and we added a graphics capability called FlyThru to CATIA to enable people to actually do a walkthrough of an airplane with thousands of parts. It was an incredibly revolutionary technology and it was at the heart of the 777 design, so we had a really big impact on that design.

HAIGH: So would you say that Boeing through the '80s and '90s was ahead of its competitors in the use of these technologies?

ERISMAN: Yes, I would. I think they made a strong commitment to going digital, and the 777 is widely regarded as the first all digitally designed airplane, and our organization had the privilege of being in the middle of that. It was a very exciting time. The capability in visualization was really just a laboratory tool. The executives at Boeing had seen demos of CATIA and they assumed that once you put a model in electronic form you could then visualize this model. What happened was that with CATIA, because of the algorithms internal to it, once you put about 5 models into it you could no longer move, the system would freeze. And an airplane is made up of millions of parts. So we had been experimenting with this software to develop a visualization capability and when they saw this early in the 777 program they said they would have to have it. So our researchers ended up using experimental software in the design of the 777 as a fundamental design tool. Our researchers were running around with beepers and so on. It was really a very exciting time. We made a big impact on the program.

[Tape 3, Side A]

HAIGH: So beyond those efforts you have described to help the design process, was any of the code coming out of the library, or the other projects that your group was responsible for, making its way into embedded systems?

ERISMAN: Yes, in lots of different ways. We took a step back from the company and we asked ourselves, "where does mathematical software in general, and technology also, come to play in an airplane problem?" We have all been talking about design so we said, "Let's try and understand more about manufacturing." That's when we got involved in modeling the riveting process; we built some very large simulation models that allowed you to model the whole airplane manufacturing sector. In fact, one of our mathematicians was involved in trying to optimize the assembly process. Boeing got in trouble when they tried to ramp up there production line to rapidly and started to had parts shortages and were close to shutting down the line. So they called in one of our mathematicians who brought some optimization capability that he had been working on to bear, and the attributed his work to saving the company hundreds of millions of dollars. So it was kind of right at the center of the work associated with the assembly line.

We also got involved in looking at customer support, which you might say, "Well, that's not very interesting," maintenance manuals, maintenance of airplanes, and so on. Turns out there is some incredibly interesting problems in graphics in trying to be able to turn the drawing into something alive. You get a drawing of an electronic system within an airplane, and these are huge drawings and very complicated, and by creating these visually we were able to turn these into a simulation. So you could click on them with a mouse on a particular line and you could trace it's whole path through a very complex diagram, you could click a fuse and understand what part of that circuit that turned off and on. All of this was taking this dead paper, which was part of the maintenance manuals, and bringing it alive electronically. That was less involved with mathematics, more involved with graphics, data management, what is generally called artificial intelligence, being able to automatically recognize a call out and be able to identify where that would be in the document and create links from one place to another in a document. We even experimented with artificial intelligence in wearable documents, where the documents would be available to people in their eyeglasses from a computer worn around their belt, and they could navigate through this document with speech. All of this was done experimentally in the 1990s, and much of it has made its way into Boeing products today.

HAIGH: Now during this period of the 1980s and '90s, Boeing finished up consolidating I think pretty much the entire large civilian plane marked and a large chunk of the military market. Did those mergers and reorganizations pose any challenges or opportunities for your group?

ERISMAN: They were huge challenges, and in my view they continue today as huge challenges. When you do a merger of two organizations, it's one thing to look at the products your're offering and say, "How do we put the products together?" It's another thing to look at the balance sheet and say, "How do we make money?" and that's where most people concentrate their effort. But the hardest part is to look at the culture and determine how those cultures fit together. I view a merger as being very much like a heart transplant in a person's body. The antibodies want to kill this transplant, and you have to do certain things to enable this to function. For our organization there were little spots of organizations that did some things similar to ours in other parts and we had to kind of sort all that out. There were organizations that didn't think that there was any reason to do computing. I remember one discussion with a senior executive about the strategic value of computing and thinking about how it will transform your company. His response to me was, "We ought to outsource all that crap." He wasn't thinking strategically how IT, mathematics (the math and technology piece that I was running) would really transform the company. I really spent a large portion of my time in the last four years after the merger traveling to the different parts of the new company—Los Angeles, St. Louis, Philadelphia, Arizona—getting to know the executives there, trying to talk about the strategic value of the things that we were doing, and looking at ways we could make this technology really work for the company.

HAIGH: When you say “the merger” do you mean the 1997 merger with McDonnell-Douglas?

ERISMAN: McDonnell-Douglas, Rockwell, Hughes—they all kind of happened simultaneously in the late ’96, ’97, ’98, that time period. It was in that time period also, really a bit earlier than that maybe, that Boeing Computer Services became a part of something called the Shared Services Group, and the Shared Services Group in Boeing was managing the computing infrastructure, managing the logistics, trucking, buying power, the health clinics, all the things that were done in common across the Boeing company, and what was left of computing services became a part of that. So my organization, this technology organization, reported into Shared Services then until after the merger. McDonnell-Douglas had had a group which they called the Phantom Works, which was an R&D center for McDonnell-Douglas. Boeing didn’t really have an R&D center until that time. What Boeing had was a Vice President of Research who oversaw research programs in the different divisions, but these programs were division specific. Ours in computing was company wide, but we were located in the Shared Services division. So after the merger with McDonnell-Douglas, the new Boeing looked at us and said “this organization ought to become a part of the Phantom Works.” So around 1998 I moved from reporting into the Director of Computing and Shared Services to reporting to the President of the Phantom Works, who was living in St. Louis. So I ended up making many, many more trips to St. Louis than I had ever made before in my life. Ultimately he moved out to Seattle as President of Phantom Works and I reported to him until I retired in 2001.

HAIGH: So did that official relocation into more of a research division have any impact on the actual work of your group?

ERISMAN: Yes and no. I would say it did in the sense that we got involved. There were some projects that were formed within the R&D group that went across all of the divisions, so we became a part of some much bigger projects; some of our people were put on teams that were a part of these bigger projects.

The whole planning process for research became very bureaucratic. From 1990 until ’97 or so after this merger and until 1998 or so when we became a part of the Phantom Works, we had a significant amount of R&D money that we could work with, and that was a real breakthrough for the organization. We accomplished some great things that I think are still impacting the Boeing Company, whether it is the assembly line or FlyThru or other things. I mean there were some really great things that happened. After we became a part of the Phantom Works they developed a common system by which they determined which projects ought to be funded, and it was very bureaucratic. It would start almost in April for the next year. It was painful, you constantly had to prove yourself, no matter what you did one year you had to prove, well can I do it again this year? We were measured in terms of the amount of value that was generated for the company as measured by them, which was a good thing and our organization always came out really well by those evaluations. But it was a painful bureaucratic process is all I can say, so it certainly affected us in that way as well.

HAIGH: Now would you say that there were some functions or businesses within Boeing that were more responsive to your message than others were?

ERISMAN: Certainly true, and this happened in almost individual ways. We built this incredible relationship with the people in Customer Support in the commercial airplanes, and they saw that the kind of thing we produced for them created great value for them, whether it was electronic documents, wearable maintenance procedures, or whatever. So we had very high level strategic meetings with these peoples at their Senior Vice Presidential level and doing the planning. We were greatly a part of their team.

We had some resistance in the manufacturing area. Hal Scott was the mathematician who did the work on optimizing the flow in the assembly line. He made a presentation to a group of senior vice presidents in our building talking about this idea and they started arguing about whether their way was better and a true argument broke out where we had to call time out. The most senior vice president came up to me and he put his arm on my shoulder, I will never forget it, he said, "I guess we are not ready for your technology. We have a few other things to do." We ended up putting that technology on the shelf for four years. And four years later when they had a problem in the assembly area that was really going to cost them a lot of money, I got a call from someone in that office who remembered our technology and said, "Could you help us?" This mathematician and I jumped in the car, drove to Renton, and were in this guy's office 15 minutes later. This was probably eight years ago, and as far as I know he is still working with them in adapting solutions to create a great flow in the factory. So we developed a very important relationship with that group.

There was an aerodynamics group in the commercial airplane company, and they recognized that they had great mathematicians who were aerodynamicists, but we brought a great perspective to that table. We started working with them probably in 1974, and we have had a very close working relationship with that group and I'm sure it continues today. There were pockets that were very, very close and there were pockets that were almost impossible to penetrate where there was huge opportunity, but just nothing to make it work.

HAIGH: So from that example you gave, it sounds that the problem was not that they didn't see the potential of technology, but that they preferred to do it themselves in their own way.

ERISMAN: Sometimes it was that way and sometimes they just didn't see the potential. I know one of the issues associated with exploring design space. The engineers had a way that they used to do this which was mainly brute force, and they continued to try and say "this is the way they ought to do it," even when we presented them a tool, which I think we called Design Explorer, that allowed them to explore regions of this design space that they would never get to. We would present them with options, and then they would take those options and use their old hand methods and say, "This is what we did," as opposed to accepting the role of the technology. So there were some people that said, "We don't need it"; there were some people that said, "We would rather do it ourselves." This is always going to be the case in a large organization. As a result, I think my last five years at Boeing was primarily devoted to breaking down those barriers to allow our organization to help the company.

HAIGH: So when you were talking about the '70s, you mentioned training sessions and newsletters, those kinds of outreach efforts directly at potential users of the library. How did those kinds of efforts develop through the '80s and '90s?

ERISMAN: Well, the newsletter continued to be published as a hard copy, and then I think it went on the web in the late 1990s so it was published electronically. I don't know that it is still published anymore, but it was published as long as I was there. We continued to do weekly seminars. We got to the point where we would have a seminar every week around different areas of technology, and our goal in that seminar was to be able to communicate to potential customers, and then we would invite customers that have particular interest in this topic. So every week there was a meeting going on that would bring people in to the building and give them an exposure. Sometimes we would do seminars on customer sites as well to introduce them to technology. We did executive seminars where we brought executives from around the company together on a quarterly basis to show them new possibilities from technology and how this could affect the company.

HAIGH: You've already talked about the impact of personal computers and workstations. Thinking about the late '80s and '90s, I would imagine that two other technologies that might have had an influence would be the proliferation of networks and then the Internet, and also (on the higher end) the rise of massively parallel machines.

ERISMAN: Right. I would say certainly networking changed the nature of computing. I think about the idea of being attached to a 19.2 kilobit line and how that has changed over time and continued to change. Wireless has created an opportunity for mobility that has changed it again. So yes, there is no question that that has had huge impact on the way we did our work. I remember the first wireless network we set up. I had a networking component in the technology organization, and we set up a wireless network just to experiment and to try things. We found that people could get access to the Boeing proprietary network by sitting in the parking lot, and realized we had some big security issues to deal with.

The Internet, I will never forget the day one Ray Allis walked into my office at 6:15 in the evening and he said, "I've got something I need to show you." We went down and he had found Mosaic, this is probably 1993, which enabled you to graphically go out and find all sorts of information on the Internet. It was like magic. This was a dream that everyone in computing had had for years, that all this information is out there and you can go out and access it, but you really couldn't until the Browser. We held seminars across the company. Ray became known as the father of the Boeing Internet because he kind of led the evangelistic effort to change the way Boeing used the Internet. Boeing now sells all its parts through the Internet and parts supplies, they work with suppliers. The Internet transformed the way the company did business. It was huge.

With massively parallel systems, the potential was always there, and to my knowledge while I was at Boeing, we did a lot of interesting work from an R&D point of view. Our mathematicians looked at algorithms; they looked applications; we did experimental things with massively parallel systems; we got involved with grid computing, which is using the whole grid as a computer and using the resources of various areas. A fellow on my staff, Ken Neves, led our activity on grid computing, along with Tom Wicks. It was great opportunity. The extent to which this became real in Boeing is still a question in my mind. It was certainly real for the research community, but I don't know that it's yet fully real for the average person in the company. Whereas the Internet affected everyone, networking effected everyone, even vector computing affected everyone in scientific computing; but parallel computing less so I would say.

There is one other important factor about parallel computing. With scalar and vector computing, you could clearly say that a speed up in the computational algorithm would translate directly to a speedup in the solution of the overall problem. This is not quite true with parallel computers. What you can do in parallel with the algorithm within the overall code opens up all sorts of new possibilities. You can't assume all of the parallel resources of the machine should go to performing the computations on the algorithm. So the paradigm here is really different, and I believe the role of libraries here is still being worked out.

HAIGH: I saw on your resume that you were a member of the Boeing CIO Council.

ERISMAN: Right.

HAIGH: So what was that group?

ERISMAN: This came about right after Art Hitsman had left around '95 I think, and a fellow named Terry Milholland came in, and he was formalized as The Boeing Company's Chief Information Officer, CIO. He didn't have all the resources of computing reporting to him from around the company, so what he did is he created a CIO Council and this council represented people from the military division, the commercial division, as well as the data center, and he just wanted to look at computing issues in common across the company. He decided that we needed a forward look in addition to looking at the operational issues and asked me to join as a member of the CIO Council. After the merger, the CIO Council grew in importance because it took in the computing activity that was going on in the former McDonnell-Douglas in St. Louis and all the other places. So the heads of computing from all the major divisions of The Boeing Company, so there were eight people who were part of the CIO Council and I was a part of the CIO Council from an R&D point of view. So we would meet on a regular basis to talk about issues in technology. Terry in particular encouraged us and encouraged me to communicate new ideas to these people that would cause them to think about their work differently. I pushed pretty hard on some issues as wireless started coming in. Certainly we pushed out the Internet activity and so on. So it was a neat opportunity.

HAIGH: On that kind of managerial level, how much do you think your scientific computing operations had in common with administrative data processing?

ERISMAN: Well, the CIOs went beyond administrative data processing. Furthermore, though, by that time my role had changed. I was not only involved in scientific computing. As a person responsible for the R&D and technology, I had a group that was working at issues of security, of telecommunications, data management. So in those senses, the broad areas of technology were very much involved. It's true the issues of scientific computing were less involved in that activity, so my role had to shift a bit to be involved in those other areas of computing.

HAIGH: In 2001 you retired from Boeing. Why was that?

ERISMAN: Can I catch one thing from earlier?

HAIGH: Sure.

ERISMAN: I will have to fill in the detail later, but I think this was an important event. For one year the president of Boeing Computer Services was a person named Mike Hallman. He was the president right before Art Hitsman so it must have been 1989 Mike Hallman came in. Hallman came in and he had a meeting with Cray, and the people from Cray were incredibly arrogant at this meeting. It was just one of these meetings that I will not forget. John Rollwagon had left at that time and his replacement was there and it was just a very arrogant person. Mike Hallman said we need to find a way to work with someone other than Cray in our scientific computing work, and Steve Chen had started a new supercomputer company in Wisconsin; he was the architect of the XMP and the YMP. So he got some funding from IBM and others and started a new company, and Boeing actually bought a share of that company and I was on their advisory council. I remember telling the Boeing Company that they shouldn't invest in this company because I didn't think it would go anywhere, but there was very strong motivation to do so to get an alternative to Cray. So since they decided to invest, I said, "Well, if we are going to invest in this company, then we should invest in the ability to use this technology, so let's match dollar for dollar in investment so that Boeing can do some R&D work." That one year working with Steve Chen's company, the company failed, IBM pulled the plug. But we had a year or two which produced great results in another way for Boeing. Not from the point of view of a great commercial product, but because we were investing in how would we effectively use this tool if we got it. In that investment we produced the FlyThru that I mentioned that was a part of the 777. We produced several other things that were very, very innovative and in the end I would say Boeing lost if they thought they were investing in a company that was going to do well, but they won from the investment that they made in technology, so it ended up being a good investment for the company.

All right, let me go to 2001. In 1996, a friend of mine from outside the Boeing Company, an ethics professor, and I sat down and had a conversation. He was a critic of technology because he said technology is the factor that is creating all the pain in the world in addition to all the good in the world—technology is this ambivalent and powerful force. He said, "I have been against technology for a long time and you are obviously for it, and we are friends, do we have anything to say to each other?" And we started a conversation. We went off and taught a class at a Christian university, and it was a special class in the summer, it was called Business, Technology, and Values. It seems like a strange combination, but we were exploring how technology was transforming the way people work and what this does to underlying values and ethical questions. It turns out that in that exploration, we got a lot of people very excited about what we were doing and we launched a magazine in 1998 called *Ethix*. This was launched in October of 1998, built on the proposition that as technology changes the business world, there are new ethical questions you need to think about.

Now, this relates to my technology job obviously because we were looking at technology as the force. But it relates in another way that I didn't fully appreciate at the time. One of the difficulties in a large company like Boeing, and probably any large company, is you get a new idea in technology, but the company wants to resist this, and they resist it because they naturally resist change. In fact, I have drawn the analogy between the body's immune system and a corporation's culture. They resist the technology because they want to preserve life as it was, and you see this happening everywhere. The other side of this, though, is that corporate culture is also a critical part in preserving the ethical stance of a company. If a company says we don't do things that way here this is a part of their culture and you reject bad behavior. So there was a tight relationship between technology, corporate culture, transformation of technology, and business ethics. At least there is in my mind.

So in '98 we started publishing this document. I got the approval of the Boeing company to put this out, and while I was still a part of the Boeing company I did this on my own time, so this was a weekend project that I would do and during the week I did my job as R&D director for technology. It became apparent by about the year 2000 that I was going to have to make a choice. For some people it would be time to retire. I was not interested in retiring, but I really couldn't carry on in doing what I was doing with *Ethix* and a growing circulation list and a growing opportunity to do speaking and discussing this area and also do justice to my job at Boeing. So in May of 2000 I made the decision, and I went to my boss and I said, "You know, I have done this for a while, and I am really tired." It was true; it was a hard job that I had. I mean, I loved what I did, but it was a very hard job in keeping the organization sold, keeping all these people happy, building an organization that had some really great people in it, and it was just hard and I was tired and I couldn't do both of these things. So I said I think I'm going to retire and I think I will probably do it in the first of July in 2000. He convinced me that we should take a little longer than that and do an orderly transition, so after some negotiations I agreed to stay until April of 2001.

So the rest of 2000 I was working on a transition plan and getting a person to replace me. She actually took over the organization in January of 2001 and I went off and did a special assignment and continued as consulting support to the organization, but I actually moved out of my office and moved down the hall so that I could give her the ability to take over the organization. And then I left in April of 2001. It was really interesting, I had done this work for 32 years at Boeing, and the next day I woke up and it didn't seem unnatural not to go to work at Boeing, and I have never looked back. So it was a great transition for me.

HAIGH: So you would say then the decision was motivated completely by your growing interest in these ethical matters rather than any dissatisfaction or unhappiness with the way things were going at Boeing?

ERISMAN: Absolutely. I would also say it was motivated by being tired. It was a very hard job and a very large company to keep this sold. I would say that motivated me on the one end. But I would say it was primarily the desire to do something else. I also started teaching. I had said very early in the interview that I came to Boeing with the idea of spending five years there and then going to a university. Instead of going to a university to teach mathematics I ended up at a university teaching in the business school. Because this whole issue of technology transformation of business, technology transition are all business questions, so I ended up teaching at a business school on business technology and business ethics. So right now I am doing about half time teaching and the other half putting out this magazine and I am speaking really literally all over the world on the subject of business ethics.

HAIGH: Would you say that prior to 1998 you had had any personal commitment to integrating ethical issues with your own managerial career?

ERISMAN: Yes, I would say that I always had that. I used to call not just for ethics for damage control, not just ethics to stay out of trouble, but ethics for mission control, which is creating a great organization. Believe me, our organization was far from perfect and I had a lot of weaknesses as a manager and I can look back now and see them. But we really worked hard at creating an organization that was fun to be a part of, that acknowledged the value that people brought to the table, that appreciated people, and that valued their creativity, and that's a part of ethics. So yes, I would say that that is a part of it.

I think it probably goes all the way back to my childhood. I was raised as a Christian and for years when I was younger my mother would read to us from Proverbs. I don't know if you have ever read the book of Proverbs in the Bible, but it's filled with examples of doing business in a just way and treating people well and responsibly—just these little pithy sayings over and over again about how to treat people. That became a part of who I was all of my time at Boeing. So in that sense, yes, this was nothing new; it was a part of who I was.

HAIGH: How successful would you say the magazine has been?

ERISMAN: I would say that it is marginally successful. I get some enthusiastic readership. We distribute about 2,500 to 3,000 copies around the world. The contents of it we give away, which might undermine its sales because it's out on the web, and we get 50,000 to 70,000 hits a month on our website from all over the world. 4,000 hits from Poland last month, which is kind of interesting. So in that sense it has been successful. I hope that it has been successful in trying to create a picture for people that ethics is more than just staying out of trouble, that acting ethically in a world changed by technology means thinking more than how do I find a new loophole. I don't know if you have read any of the stuff on Enron, but some of the books are absolutely fascinating. *Conspiracy of Fools* and *The Smartest Guys in the Room* are insightful books, showing they just did everything they could to manipulate numbers and totally missed what business is all about. There was a total lack of integrity.

I guess for me I have always looked for a sense of wholeness in my life, not my family life in one corner, my church life in another corner, and my work life in another corner, and you operate by different rules in each one, but how do you be a whole person. Being a whole person has always been an important part of my life and my way of thinking. And I suppose in some ways that kind of mirrored my career as well, because I started working thinking I would go teach, but I wanted to get a bigger context on mathematics so I wanted to go to industry. In going to industry it was a really good thing, but I wanted to see a bigger picture of mathematics, so I ended up managing the mathematics organization. I was interested in understanding the technology context in which mathematics fit, so I ended up managing that organization. I ended up working across manufacturing and so on. So this idea of wholeness and context is something that has always driven me. And I think it is reflected in the way I've seen math libraries, is that a math library is a tool; the mathematical software needs a context in which to operate; how does it solve the problems that people really have; how do you communicate with users to meet real needs, and so on and so forth. I don't know, it's all one package in my head.

HAIGH: While we are dealing with the most recent stage of your career, you mentioned on your resume that you are also doing or have been doing consulting work for a number of clients.

ERISMAN: Yes. I ended up doing some speaking on business ethics. It was prompted by the magazine *Ethix* so I suppose that would be another way you could measure its success, and in that way it's been successful. I have had some wonderful opportunities. I have spoken to a business audience in Finland; I was a guest at the UBS leadership center—a wonderful experience in Switzerland. Last fall I was in Beijing, and had an opportunity to address a large audience of people to talk about business ethics in an age of technology, sponsored by the government of China, led by the person who is on the Council of State Owned Assets for China, a very senior person in the Communist Party. I spent a week doing lectures in Hong Kong last march. I am leaving for Jakarta in two weeks and Singapore and Malaysia. [In June 2006 I went to the Central African Republic to investigate what it would take to do economic development in a country that is hurting.] So it's really been a very rich opportunity. I find that when I do these things I learn a great deal gaining a context from a different culture. I also do some consulting for other companies. I am meeting with a group of doctors at an off-site retreat taking them through some ethical discussions. I met with a small company recently and taking them through some ethics and technology discussions. It's just great fun.

HAIGH: So the consulting work and the speaking are really very tightly coupled.

ERISMAN: Very tightly coupled and very tightly coupled also with the magazine. I have had the opportunity through this magazine to get introduced to some very prominent people. I have interviewed the CEO of the Singapore Exchange; I interviewed John Reid, who was the CEO of Citigroup; I interviewed recently the CEO of Starbucks, and just sat down and talked to them about how technology is changing their business, how values and ethics shape the way they do business, and so on. I'm having a lot of fun doing the things I'm doing.

HAIGH: Well, it seems the main area that we still need to address is your general involvement with the mathematical software community, contributions in the area of sparse matrix software, present contribution in conferences, and involvement in various blue ribbon panels and groups. Perhaps we should take a short break before that.

[Tape 3, Side B]

HAIGH: Yesterday you mentioned attending a local SIAM chapter meeting early in your career had led to your first exposure of this idea of software that utilized sparse matrices. I think you had already mentioned, at least briefly, some continued involvement with SIAM including serving on its council, helping to establish special interest groups and organizing a conference on power systems. I wonder if now we could move back and talk in some more detail about how your involvement with SIAM developed after attending that first chapter meeting.

ERISMAN: I told you the story of how I ultimately got the software to use at Boeing by going through the person at Cornell.

HAIGH: Yes.

ERISMAN: Because I had gone through that person, the folks at IBM had found out that I was doing some work in sparse matrices, so they invited me to a conference they held. I think it was during 1971 at Watson Research Labs. I remember the invitation came on a telegram—that shows you where technology has gone, a physical telegram inviting me to this conference. At that conference, of course, you start building a network of people because that's the way those things work, and through that I had the opportunity to go to a conference in Cambridge, England in 1972. That was a NATO sponsored conference called "Decomposition of Large-Scale Systems". We had people from all over the world there, 50 people from probably 30 nations. That's where I met John Reid, and John and I started discussing things and writing papers together. That led me to go back to Harwell in 1974 for a longer period of time; I think it was like six weeks, and Iain Duff Had just started working there. So the network of people just continued to grow.

My first SIAM meeting after that local chapter meeting, I went to the national meeting in Denver in 1970 and presented a paper. I started getting acquainted with the people at SIAM, got to know Ed Block, became a member of the Council. They added what was called a visiting lectureship program where people from industry would get called to go to universities to present a lecture, and I did that for a large number of years. I think I was Vice President of our local chapter in the Seattle area, the northwest region: Oregon and Washington. So SIAM was really a big part of my life until probably the late '80s, when with my other duties I just wasn't able to keep up. In addition to running the special power systems program for SIAM, I was the General Chairman of the SIAM national meeting in 1984 in Seattle.

HAIGH: What did that job involve?

ERISMAN: It involved organizing all of the sessions, organizing the facility, getting a program committee. We had something like 700 people here. We did one innovative thing at that meeting, though, which I don't know if they have continued. But we asked, "is there some way we could get high school math teachers involved in SIAM in order to further the context from which they are teaching mathematics and think about the applications areas?" So we organized that SIAM meeting so that the final day included high school teachers. I don't remember how many teachers we got out for that day. We had the invited lectures for SIAM for those sessions, make sure that they talked at a level that would communicate with these teachers, and then we had some special sessions during the parallel sessions for the high school teachers to talk about the teaching of mathematics. This was an idea that Ed Block had had and he pushed us toward this. It was a good idea I think. Whether it was continued in SIAM I don't know. That was 20 years ago, so I don't know what happened to it from that point of view.

HAIGH: And how would you describe Ed Block?

ERISMAN: Ed Block was a wonderfully curious, creative, very nervous, always sure that something was going to go wrong, always certain that something wouldn't get done. But I liked him a lot. We would meet either here or in Philadelphia and met a lot of times other than just the formal meetings.

HAIGH: So was SIAM something of a one-man show in those days?

ERISMAN: Well, certainly Ed was a predominant figure in SIAM in those days. But they had their councils and their committees and people took an active role. It was through that that I got to know people that I know to this day, people like Bill Gear and Cleve Moler and Gene Golub. So it was great. I really thought SIAM performed a wonderful role. I have been involved in some societies since then, and they don't have what SIAM had. I was trying to figure out what it was, and I think in part it was because SIAM was a bit smaller. You go to something like the IEEE, and you have a branch called Circuits and Systems, which is a huge branch within IEEE; or Power Systems which was also huge. I used to go to the Power Systems meetings in New York and you couldn't possibly get to know the whole crowd. It wasn't nearly as collegial just because it was so large, I would say.

HAIGH: Now during your time on the SIAM Council, were there any initiatives that you were involved in proposing or supporting? Any way that you might have left some personal stamp?

ERISMAN: I don't remember history well enough to be able to say exactly. I do know that the issue of putting more emphasis on the "I" in SIAM was a part of what I was trying to be engaged in doing. There are some others. Roger Crane from RCA, George Burns from Exxon, Wim Schafers at DuPont. These were people who also had organizations within industry that I built links with, and we tried to do more with raising the role of the "I". I was very interested in the applications area, and I think the collaborative conference we had with the power systems group might have been the first of its kind in that way. Then this whole idea of having special interest groups in numerical linear algebra and software and so on was one of the things that came out. I remember working with Bill Gear and Gil Strang on that. I don't remember whose idea it was, but I know I was involved in forming the special interest groups at SIAM.

HAIGH: Do you know what year that would have been in?

ERISMAN: I was on the Council from 1980-83, so it was probably in that timeframe.

HAIGH: So prior to that, SIAM hadn't had any SIGs at all?

ERISMAN: No. But it wasn't that it was a wholly new idea. ACM had SIGs, so we were really borrowing an idea that had already existed; it wasn't a brand new thing.

HAIGH: So leading from that, had you ever had any involvement with the ACM's SIGNUM, which seems to have been the main group in this area in the late '60s and early '70s.

ERISMAN: Yes, I was involved in some SIGNUM meetings. I gave a talk at one of the SIGNUM meetings; I used to read the magazine, the *SIGNUM Newsletter* I think it was, regularly. So I was involved to some extent. I was a member of ACM and SIAM at that time.

HAIGH: Broadening then slightly from that to discuss your involvement in the mathematical software community more generally, I know that you attended at least the second of John Rice's mathematical software conferences. Did you go to the first one?

ERISMAN: I don't think so.

HAIGH: So prior to that conference, had you known John Rice?

ERISMAN: I think we had met at a SIAM conference. There was some fixtures in those days at the SIAM conferences that everyone wanted to meet. George Forsyth, who was kind of a person I admired a lot in linear algebra; I loved his writing and I loved his speaking. There was Beresford Parlett , who I am sure he doesn't remember this, but when I gave my first contributed talk at a SIAM meeting in 1970 sat in the front row and made some very encouraging comments to me, which was really nice. At the first Sparse Matrix Conference Alston Householder made some nice comments to me which I remember. So there were some really encouraging figures there. Those are some of the people that I can remember from that time.

HAIGH: The second mathematical software conference was held at Purdue in 1974. So is your impression that by the time you went to that you would already have met most of the people at SIAM?

ERISMAN: That conference is something that comes to mind every year on my daughter's birthday. She was born on a Friday and I left on Monday for the Conference. My wife still reminds me of it! I think I had met a number of the people, yes. There was a mathematical software conference at University of Texas that happened in 1978. I think that Andrew Sherman was the program chairman, and the usual suspects were on the organizing committee, Tom Aird, Jim Cody, Stan Eisenstat from Yale, myself, Phyllis Fox, John Rice from Purdue, and Larry Shampine from Sandia. So that was a conference that took place, it must have been in the late '70s or so. I've got some correspondence here from October 1977.

HAIGH: So that thread of meetings continued?

ERISMAN: Then some IFIP activity at the one in Baden, Austria [1978] where I ran a panel discussion was another one of the software conferences.

HAIGH: My impression is that some of the energy shifted way from these ad hoc mathematical software conferences and into the international working group.

ERISMAN: Right. Then I think it was Beresford Parlett who got me on as a speaker at a AAAS meeting on why people don't use software that other people write. I think he is the one that made that connection for me.

HAIGH: So the common thread among these presentations is the role of users and the need to take them more seriously in designing packages.

ERISMAN: Right. And to recognize that this is not the mathematical community talking to each other, but this is the mathematical community making a difference for a bigger community.

HAIGH: So as you delivered that message at meeting after meeting, do you feel that it's a message that people were responsive to?

ERISMAN: Some people were, yes. I think there are some people that are very content to work at the next generation of the algorithm that they have worked on for many years, and the context of it is not relevant to them. I respect those people because I think they make a contribution to the field by doing work in very detailed areas that are ultimately important. So for those people this message probably didn't mean very much because it didn't change there agenda at all. But I think it had an impact on some people.

HAIGH: I'm assuming you probably don't want to name names in terms of the people who didn't care about users. Are there any people you can identify as kindred spirits or people who might have been influenced by your pleas to look more at users?

ERISMAN: I would have to think about that one. Certainly my colleagues at Boeing. I also found the British mathematical community more tuned into this at that time than was the American community. That's where my friendship and deep respect for John Reid started and continued.

We tried to build a whole organization at Boeing based on this kind of philosophy of thinking about the users of our services, the people that needed it, and the end results of what it is we were doing. I ended up throughout my career speaking at places also like IEEE Circuit Theory and IEEE Power Systems and a CAD conference and these kinds of things, and I often was disappointed that more of my mathematical colleagues wouldn't push out. One of the things that always impressed me is that as you read the literature of these other fields—structural analysis, power systems, circuit design, control systems —it's filled with mathematics so why doesn't mathematics play more of a role for being an integrator for all of these?

HAIGH: My impression from reading transcripts is that this fact was widely bemoaned within the mathematical software community, but people would often place more blame on the users for not seeking out the mathematics.

ERISMAN: Right, instead of saying, "Is there something that we can do to address this." I was on one of these committees put together by the National Science Foundation on examining the state of mathematics in the world. This was a fascinating committee. It was run by someone who was not a mathematician. It was populated by a very diverse group of people many of whom were also not mathematicians. There was a chemist from Harvard, a mathematician from England, a mathematician from Japan, a financial person from Wall Street who did mathematics in finance. We wrote a report for the National Science foundation that came out probably in 1998. This was a big focus of our discussion as well: is mathematics a unifying language for many of these fields? But when mathematicians don't engage with these other fields, what happens is that there comes kind of a separation, and as a result the language that gets developed within these other disciplines is no longer transferable because it isn't in a common language that people understand. It's little things, it's Ax=B in mathematics, it's solving equations. It's Yv=I in power systems, and Y is the admittance matrix and I is the input currents and V is the voltages. It's still Ax=B, yes, but the way they have developed the language around it kind of interferes with some of the potential transition that can take place, and how can we be more inclusive and more conscious of this.

HAIGH: Your resume shows that committee as being 1994 to 1998. Do you think that the committee had any influence? It presumably published a report. Do you know if that translated into any translatable progress?

ERISMAN: I don't know. Let me say two reasons that make me wonder. I'm not sure that there was growing funding within the National Science Foundation for Mathematics, and we were arguing that there was a need for more funding. We were also arguing there is a need for mathematics departments at universities to be a little more outward looking and not accept the idea that engineers ought to teach their own calculus because they don't like the rigor of mathematics, but in fact go to engineering schools and teach the calculus for them in a way that would communicate with them, but would maintain this core of knowledge. We were arguing for those kinds of things and for funding. I don't see a lot of evidence that there have been changes at universities. I don't see a lot of evidence that there is more funding.

In fact, Tom Friedman in his latest book called *The Earth is Flat*, a fascinating book, argues that because of telecommunications and dropping of trade barriers, the world is getting flatter: jobs are being outsourced, products are being made in China, telephone answering centers are in India, you don't know where they are—the whole world is getting flatter. He says, "How do you compete in a flat world?" You compete by increasing the ability of Americans to know math and science so that they can compete with new ideas in a changing world. He is bemoaning in this book the fact that there is less funding for math and science, there is less education funding, and we are just not getting there as a country. So did the report have a difference? Well I don't think so, but I think there are other voices that are kind of carrying on a similar point and perhaps it will make a difference.

HAIGH: You were involved in a number of similar committees, including National Academy committees on supercomputing in the early 1990s and one on "information technology in a competitive world" from 1996 through 1999.

ERISMAN: Right. I'm involved in another one today that I'm just finishing, a National Academy Committee on Review of the National Institute of Standards and Technology. They put together this committee and then we look at the NIST programs.

HAIGH: So can you talk through each of those perhaps in terms of the objectives, what the issues were, and any personal contribution you think you might have made that would have influenced the group's final findings.

ERISMAN: Well, if we go back to the NSF committee for a minute, there is a diagram on our final report that shows this integration picture and a possibility of fragmentation of mathematics if it doesn't play a stronger role in this overall community. I think the contribution I made in that committee was dealing with that particular subject. In the supercomputing committee, what we were looking at the role of supercomputing going out into the future? It was kind of uncertain, Cray was kind of on its way down, and the question was "where is supercomputing going, where are the real problems?" We tried to look at the issues of how do people attack the next generation problems and what is the need for supercomputers to help make this happen? It was a fun committee to be on. I remember we said at the time it would be good five years from then to look at it again and see if we made a difference; as far as I know that didn't happen.

Making IT Better was a book that came out of the IT committee that I was on, and I think the contribution that the committee made and that the book made (and I know that I contributed to it but I would not want to say what role I had in it) was looking at IT in its broader social context. We spent a lot of time talking about the fact that technology can be a force for both good and bad. It can create opportunity and lead to outsourcing of jobs, it can lead to the rich getting richer and the poor getting poorer or it can lead to more opportunities. Technology is a dual sided thing, and we spent a lot of time talking about that and it shows in our report. That was a fun committee, and I think we made a contribution.

With the Committee for Review of NIST I spent a couple of years looking at the mathematics program in NIST. And incidentally, from a mathematical software point of view NIST is a major player and often isn't mentioned in these things. They put out that first handbook, *The Handbook Of Mathematical Functions*, which has been a staple for a long, long time in terms of mathematical functions. But then they are now putting out a new version of that handbook which will now be involved with the web and have software with it, so you will be able to not only look at this function but you will be able to do some computing with that function. I think that is coming out pretty soon. I don't remember exactly the date, but it's coming out pretty soon. The last couple years I have been chairman of the IT laboratory review team, which includes their mathematics as well as the other pieces there, and we are right now in the process of putting out our report for how NIST is doing. This kind of seems to keep coming full circle in my life, but because of issues like homeland security, the World Trade Center, the issues of security in the Internet, these people at NIST are getting pushed into broader and broader objective of which their technology is a key to solving a problem that is bigger than they are. So it's forcing them out beyond the details of their algorithms and forcing them to look at things in a bigger context. So anyway, our report will come out at the end of this year and that will end my involvement in that committee; six years is the longest time one can serve on this committee I have been involved for seven years due to some transition issues.

HAIGH: So collectively those committees represent quite an investment of time on your part.

ERISMAN: Right.

HAIGH: Do you think that there is anything that makes you particularly valuable as a member of these committees or just particularly interested in volunteering for them?

ERISMAN: I do it for the money [laughs]. They are all volunteer committees so we don't get paid. I think maybe what I bring to the table there is my experience at Boeing, which is Boeing is a big user of mathematics and technology. I had a position where I had an oversight, so maybe that's what I bring to it.

What I get from it, though, is certainly much better. I think that the interaction with people and the interaction with the scientists at the laboratory, the interaction with the development of these reports has been an incredible opportunity to learn from others. One of the things I worried about when I left Boeing is if I am going to be involved in business, technology, and ethics, and I am no longer running a technology organization, how do I keep a link to technology, because technology is a rapidly changing world? So through this NIST committee I have been able to do that and it's really been a wonderful opportunity of staying in touch with the technology. I have also been chairing a board at the Washington Technology Center, which is a state managed investment in creating jobs in the state of Washington through technology. We give funding to allow university researchers who have created new technology to work with a company to create new products which lead to new jobs at the state, and I've been chairing that board for the last couple of years as well. So that also creates another link to a world of technology.

HAIGH: So unless you have anything else to say about those panels, it might be appropriate now to focus in again and talk specifically about your involvement with sparse matrices. So you've told the story of how you got the original piece of IBM code from Cornell, and of putting that to work within Boeing, but clearly that wasn't the end of your interest in this area. So how did things develop after that?

ERISMAN: Well, I think that the sparse matrix work in the early days was really very exciting because it was a wide open field that didn't have a lot of definition to it. It had a lot of practitioners involved. There were a lot of conferences because sparse matrices were not mentioned too much at SIAM or other places because it was just a small field. A community grew up around the sparse matrix area. We had conferences every year or two that usually resulted in books, and I ended up writing chapters in quite a number of these books. There were a couple of conferences in England, there was one in Tennessee, there was one in New York at IBM, there was one at Tromso, Norway, and one in Austria. These were great opportunities with people that became friends and to push forward new ideas. My involvement in it initially was driven by the problem I had to solve in the circuit analysis, but it ultimately became an interesting area of study and I got very interested in seeing how this worked its way out in chemical engineering and structural analysis, in linear programming, and distribution logistics management. Each of these had some sparse matrix implications. So writing papers, being a part of that community for a number of years, and ultimately writing the book which came out in 1986, was an incredible opportunity and I enjoyed it a great deal. I have had much less that I have been able to do in the sparse matrix area since then because of my other responsibilities, but it was an area that I really enjoyed working in.

HAIGH: So would you say then that your main contribution to the community had been in terms of looking at application areas for sparse matrices, or were you coming up with new methods or presentations on a technical level?

ERISMAN: There were a number of papers that I published that also introduced new ideas into the field. I was, after all, a mathematician and I did explore some of those things. Often they were driven by particular requirements from an application, though, and that's what caused me to look in areas that maybe other people weren't looking in. I remember very early on when we were solving these very large systems of equations, and the challenge I put in front of people was nobody is doing any error analysis. How do you know what you are getting out is anything worthwhile? So I remember trying to get funding to do that work and I was trying to get funding from the customer, ultimately the Air Force customer that was paying us to do this, and they weren't interested in funding error analysis for sparse matrices.

But I spent a lot of time wondering if these simulations really mean anything. Do these numbers have any semblance of reality? Ultimately, I said, "You know, we don't even know this data precisely, and there are all of these rounding errors made in the computation. How do you know that the results are worth anything?" Ultimately they came back to me and said, "Would you do some work on evaluating the answers to these simulations subject to the uncertainty in the data?" I said, "Absolutely," and I worked on error analysis. I did a paper with John Reid that was published in ACM on some work that we did on estimating errors, published in *Numerische Mathematic*.

HAIGH: That's the one I am seeing, so the citation for that is "Monitoring the Stability of the Triangular Factorization of the Sparse Matrix". *Numerische Mathematic*, 1975, pages 183 to 186 with J.K. Reid.

ERISMAN: Right, so that was one of the areas where I think I made a contribution. In another area, there was a lot of controversy. Many of the sparse matrix methods are ad hoc. The problems are such that you can't necessarily prove this algorithm is optimal. Alan George did some really groundbreaking work in the symmetric case of solving sparse systems where he proved that, for certain kinds of regular grids, the nested dissection algorithm was optimal. But for irregular grids, and nonsymmetric matrices in particular, people were using ad hoc algorithms to order the equations to try and reduce the computation.

So I remember when it occurred to me to ask the question: “if we think minimum degree is the best algorithm (it’s a very local algorithm where you choose as you pivot the one that will produce the least arithmetic at the next stage) and apply that to the grid that Alan George showed where he got an optimal algorithm, what would happen?” Would minimum degree be optimum? Here is the interesting thing about that problem. The regular grids that Alan George applied his algorithm to had lots and lots of ties from the point of view of an algorithm like minimum degree, that would pick the pivot that introduced the least amount of arithmetic at the next stage. So I said, “what would happen if you play with the tie breaking scheme for pivoting, what kind of range of net computation would you get from this minimum degree algorithm?” By experimenting with some really wild original numberings, I came up with an example that showed here’s the amount of arithmetic you do using minimum degree and here’s a range. It starts from here and goes to this range. It was all experimental work. This one along the bottom corresponded to Alan George’s result, but this one was significantly higher.

It was only experimental work, and I presented this at a sparse matrix conference in Chicago. One person who I won’t name at this point came out of his chair. He had been committed to saying that the minimum degree algorithm is optimal, and what I was showing was an indication that it probably wasn’t. He came out of his chair and he was very, very negative about the result. He said I was just graphing this thing incorrectly to try and give an impression that wasn’t there. I said, “Look at the data.” That was the paper that we ultimately published with SIAM called “On George’s Nested Dissection.” ["On George's Nested Dissection Algorithm," *SIAM Journal on Numerical Analysis*, 13, 1976, pp. 686-695, with I. S. Duff and J. K. Reid].

That was one of many tracks that I ended up going down as I was writing the book that ended up in papers and the book kept getting delayed. But that was a contribution, I don’t know if it was very profound, but it was very profound for me and it was very exciting to do and fun. I think in some ways I did contribute a bit to the technology part as well as the applications part.

HAIGH: So I think you had told me over lunch yesterday that work on the book began in 1974 and it came out eventually in 1986. [*Direct Methods for Sparse Matrices*; Oxford University Press, 1986 (jointly with Iain S. Duff and John K. Reid)].

ERISMAN: Right. John and I had the idea for the book shortly after we met in 1972 at the Cambridge conference, and we signed a contract with Oxford University Press probably in ’74 when I was over there and they were expecting the book in a couple of years. The field at that time was very unstable, and every time we got together to work on the book we would look at a new idea, and that would result in a different paper or a different way of thinking about it, but we weren’t making much progress with the book. Every time I would go to England we would go meet with the Oxford Press people, they would take us to lunch (they were very nice to us) and they would encourage us to finally get this thing done. So ultimately I remember the last parts of it, we brought Iain Duff in to help us finish because we weren’t getting done, and he had had much more detailed involvement with some of the software. There were a couple of key chapters that he did the first draft of and we brought him in as a full author. We would meet in hotels, Pittsburgh, Boston, somewhere in England for weekends when we were traveling and pound away on this thing. These were the days before we could easily move things back and forth with email, so it was harder that way. But it was a wonderful project. During the project our families got to know each other and we made friends, so it was a great experience.

HAIGH: So that 12-year process didn't strain relations among the collaborators?

ERISMAN: No it didn't. These are a couple of really great guys and I have a tremendous respect for them. I think we all brought something a little different to the table. One of the signs of a collaboration, we were even able to joke about this, is that you really do have a collaboration when in the end each person thinks they had the major role [laughs], and we really did have a collaboration. Nancy and I were in England two years ago and got together with all of them and had a great time. They are still good friends.

HAIGH: So what was the perceived niche for the book? Was it intended to be the first solid book on the topic?

ERISMAN: Right. There were conference proceedings before that, but there was no book written by an author that would reduce all of the concepts in some coherent fashion. In our procrastination, Alan George came out with a book before we did. It was a different kind of book because he was focused on the symmetric problem and we were looking at the general problem.

The book came out in 1986. The fact that it's still selling in 2005 is a bit staggering to me. It's really out of date, and that's why we're doing the update. But Oxford has been very supportive of this still, so hopefully we will get an update out and hopefully it won't take another 12 years.

HAIGH: So is it a book that might be assigned for a Ph.D. student?

ERISMAN: Yes, it's a book that has been used in graduate schools. I can see citations to it on the web and so on. It is also a great book for a person to just read in the evening. It's a page turner! [laughs]

HAIGH: How is work progressing on the second edition?

ERISMAN: As of this particular date we have gone through chapter eight of twelve. I believe that nine and ten are going to be much harder and it's something that John and Iain are going to have to do most of the work on. They have done most of the work on the revision so far; my role has been pretty minor in terms of going through and bringing my perspective to it. But the next two chapters really do depend on them.

[Tape 4, Side A]

HAIGH: So I think that covers the book. Now while we are still on the topic of sparse matrices, as you were working on this area at the same time were responsible for the Boeing library, you must have an interesting perspective on the process by which work in this area would make its way into more mainstream software libraries.

ERISMAN: Yes, and I think some progress has been made in this area. Boeing ended up putting some software into its library on sparse matrices; the Harwell library has MA28 and it's successors that are sparse matrix software; there is some stuff out of Waterloo, SPARSEPACK, which is software in the symmetric sparse matrix area. So I think that yes, it has made its way into more standard computation.

What ended up happening at Boeing as I ended up bringing in a couple of people who worked in this linear algebra area, and there came a time when primarily because of my management responsibilities in the R&D organization I was no longer to contribute to that field directly. We brought Roger Grimes in from University of Texas and we brought John Lewis in from Johns Hopkins University, and the two of them along with Cleve Ashcraft and a few others carried on the sparse matrix work of Boeing. I don't know what the state of that is today at Boeing, but there continued to be some work there with some very good research people.

HAIGH: I should refer listeners who are interested in software in this area in general to Iain Duff's chapter in Wayne Cowell's 1984 book which is I think a quite comprehensive survey of what was around at that point. [Duff, I. S. (1984). A survey of sparse matrix software. *Sources and Development of Mathematical Software.* Edited by W.R. Cowell. Prentice-Hall, Englewood Cliffs, New Jersey, 165-199].

ERISMAN: Yes, that was what around 20 years ago, and certainly since then my guess is that Iain Duff does a lot of good stuff in terms of putting out surveys of what's available, so he probably has something more recent.

HAIGH: Yes. And I believe that sparse matrix support has made its way into MATLAB.

ERISMAN: I remember, just for fun I was asked to give an after dinner talk at a NASA conference on mathematical software, and I really hadn't worked in this area in a long time. So I made it kind of a fun talk. I created a little limerick around LINPACK doing all these great things, and, "It handles problems with ease. But if the matrix is sparse you are out on your arse, because it doesn't handle these." It was always interesting to me how all the effort at the DOE labs, LINPACK and its successor and EISPACK and it's follow on efforts, didn't really address the sparse matrix problem.

HAIGH: Was that related to the nature of the problems that they were working on?

ERISMAN: I think it was related to the interests of the researchers probably, as much as anything else.

HAIGH: So there is nothing about, say simulating an atomic explosion that would mean that sparse matrices would be less likely to arise in that area?

ERISMAN: No, they arise everywhere.

HAIGH: Did any support for sparse matrices make its way into level one of the Boeing library during the time you were responsible for it?

ERISMAN: I don't think so. I think it remained a more specialized capability that was a part of level two. This was more directly the responsibility of Ken Neves and then Ivor Philips.

HAIGH: So then clearly it's not something that you were pushing the library people to include as a result of your personal interests?

ERISMAN: Absolutely not. My interest was not in pushing that kind of an agenda. It became their call on those kinds of issues. Obviously when I thought there was something that ought to be done, I would say, "Here is what I think ought to be done," but it was more in terms of a suggestion rather than an imposition.

HAIGH: It also seems like this is an area where there clearly would be a large number of users potentially with problems that would benefit from it. But at the same time they would, as I understand it, have to realize that what they had was a sparse matrix, find a different set of software that would work with it and educate themselves about it.

ERISMAN: Yes. One of the things, though, that I did come to realize, a general purpose library software can do a reasonably efficient job in solving a general problem, but there are no general problems—every problem is specific. The problem characteristics of these specific problems often lead to techniques that maybe are not as generalizable as you might think. It was an interesting realization. For example, the sparse matrices that I was working on for a long time had complex numbers in them and they were symmetric. From a mathematical point of view, a symmetric complex matrix is not interesting. A Hermitian matrix is interesting because if its positive definite, you can use diagonal pivots without concern for stability. But a complex symmetric matrix is not interesting mathematically. But it's interesting in the physical entity that it represents, and we were able to use properties that you could not generalize mathematically for these problems.

Similarly (and this gets away from the matrix area, but it gets into mathematical software), we did some experimentation in the late 1970s and early '80s with the solution of ordinary differential equations to solve a stability problem in a power systems environment. These are large codes that people use when they've got a fundamental change in the load that is drawing off the power system. For example, it's very hot in the Northeast and everyone turns on their air conditioner and it draws a lot of load off the power grid. How does that grid maintain stability and not black out? This is the stability problem. Well, you solve a bunch of differential equations to do this—very large sets. There is sparsity underneath these and you take advantage of that.

All the good numerical methods had variable step integration schemes in them, which meant that you change the step size of the integration based on error criteria that you use at every step of your integration. It turns out when you apply these in this transient stability problem, you could never get good performance of these stability codes. The reason is there is something about the physical nature of that code, of that problem, that meant a fixed step integration scheme that didn't have all of the overhead, could be really fast, and would be very efficient, and you could very reliably pick the step sizes. That wasn't perfectly true; we could find examples where it wasn't quite true. But what it says is the general purpose mathematical software provided a good starting point for some of these large simulations, but you really needed to understand more about the problem characteristics of the problem. Even in our sparse matrix book in 1986, at the back of that book is a gallery of pictures which I love. It pictures the sparsity pattern of zeroes and non-zeroes in from different spplications. As you look at those pictures, what you realize is there is something more that just a sparse matrix. There is something about the characteristics of the problem that have patterns, and those patterns have a lot to do with the solution.

So it is true that a lot of times the ultimate algorithm in a particular application is not a general algorithm but something very specific. I think this has always been a challenge for people in mathematical software, as you try and do something in a general way, but it doesn't do the best job in a specific context.

HAIGH: I think yesterday you had mentioned something called the Sparse BLAS.

ERISMAN: Yes.

HAIGH: I am not yet familiar with that effort. What is that?

ERISMAN: Well, the whole sparse matrix field had arisen on a supposition that was true in a scalar computer, and that is that you will reduce the time to solve the problem if you can reduce the number of multiplications you have to do in solving the matrix equations. Well, that on a vector computer is no longer true, because the way you stack the computations, the way you can put them in a vector, so that you can kind of chunk off a whole bunch of them at once, becomes much more critical in terms of reducing time. You are willing to do more operations if they have a certain character to them. So the whole trick of how do you organize the computation into vector strings became a part of the underling foundation for a good sparse matrix code on a vector computer. It was that that would be called the Sparse BLAS. That is about as far as I want to go with it, but I think it carries the idea that in the same way you think of basic linear algebra computations, you think of basic sparse linear algebra computations, and then you think about how to do that efficiently on different architectures.

HAIGH: Was that work undertaken at Boeing or was that elsewhere in the community?

ERISMAN: We did a lot of that work. There was also some work going on at Yale in this area. Stan Eisenstat and Andy Sherman were doing some work there. There were other places, too. I don't want to claim where that started.

One thing that we did start jointly between Boeing and Harwell (which was the predecessor to the Rutherford Labs where Iain and John now work) is that we recognized that this pattern of matrices is very important to their effective solution. How do you get a good set of test problems? Early in the days of sparse matrix computation people were generating random sets, placing their sparsity in random places in the matrix and then testing the algorithms. Well, there aren't any random problems in sparse matrix computation. So we (John Lewis from Boeing was the key contributor, and he worked with Ian Duff and John Reid) started the idea and created what is called the Boeing-Harwell test collection, and I think it still exists today, and it is used to test sparse matrix algorithms. We gathered examples from linear programming, from chemical engineering, from structural analysis, from power systems, and so on, representative large examples that people could use to test algorithms and compare algorithms. That was a contribution that came from Boeing and Harwell. HAIGH: Well, the questions that I have left are open-ended general reflective ones, so before I ask those I will give you an opportunity to say any other specific things about sparse matrices and your involvement in the mathematical software community.

ERISMAN: I think I have probably mentioned this. There is a collegiality that develops that goes across boundaries of organizations and nations when you work in a community like this. That is an incredible opportunity actually to be able to participate in a community like that, and it's something that I just have to reflect on with great fondness. Yes, the work was good, we enjoyed the work that we do, but I think the collegiality was a really key point also.

HAIGH: Over lunch yesterday you mentioned that in preparing for the interview you had been trying to think in general terms for the development of mathematical software and how it changed over the decades.

ERISMAN: Yes, I think the very early days when mathematical software was free through shareware and that kind of thing, there really wasn't much of an opportunity for mathematical libraries. This whole transition from free software to software that people paid for that gave rise to not just mathematical libraries, but Microsoft, for example, is an interesting phenomenon. The role of free software and open source software in taking us back to that era is an interesting thing; not to relevant for our discussion, but interesting.

So one of the transition periods was when these libraries started becoming really important in the late 1960s, early '70s, because of the fact that people started recognizing the value of software. Then there were other transition periods that came in. The transformation to a vector and ultimately to a parallel computer architecture represented a huge transition for mathematical software that I don't think has been completely solved, certainly not in a parallel way. Because if you think about any simulation code that uses mathematical software, the step required to solve a particular algorithm is often a dominant step and a barrier to being able solve a problem at all, and therefore, getting a good mathematical library routine is a key step. But in a parallel environment it's parallel, it isn't sequential, and because it isn't sequential an algorithm doesn't impact the solution of an application in exactly the same way. I think the thinking through of what that means to solving large problems in parallel computers and the role of mathematical software there is something that is still open, in my mind.

For a long time, everyone who did a simulation developed a piece of software. So there was a lot of programming and therefore the need for algorithms to make this programming more efficient as you developed an aerodynamics code and a structures code and a control systems analysis code and so on. Today, except for very specialized applications, almost everyone uses an applications package. We saw taking place very rapidly at Boeing, probably in the late '90s. Certainly in the structures community this has been true for longer than in some of the other communities at Boeing. So someone is going to use a mathematical computation, they are going to use ANSYS or Nastran; they are not going to write a program themselves. Therefore, the role of the subroutine library, the software library, is their customer is the developer of these packages, not the general user, and that changes the nature again of the role of the library. I think people are still trying to figure out exactly what the role of the library is in an era where most people buy their solutions off the shelf.

Then of course we hit a different transition period, which was moving from mainframes to PCs. And that didn't change the library so much as it changed how we measured the value of the library because in the mainframe era we got our funding from the mainframe data centers because this made their mainframe data centers more attractive, and we could measure every use of the library. Today, when everyone has their own PC or workstation, you can't get in and measure the impact in exactly the same way, you can't fund it in exactly the same way in a large company. So all of these factors have meant that mathematical software today is in a very different state than it was in the late 1970s. I think it would be useful for a group of people to reflect on what that means to libraries today versus before, and what that should mean for the next 25 years as opposed to the last 25 years. It's pretty easy to live in regret and say, "I wish the world were the way it used to be," and that's not very useful. It's more interesting to say, "How might the world be in 25 years, and what can we do through the instantiation of algorithms on computers to make that world better in the next 25 years?" That would be a fun exercise to go through, and maybe someone is going through it today, but it is something that I would certainly encourage.

HAIGH: Now I think earlier this morning you mentioned in retrospect becoming aware of some weaknesses in your approach to management earlier in your career, so I was wondering, looking back, what would say your biggest strengths and weaknesses were as a manager?

ERISMAN: I think my biggest strength was that I cared about people, I cared about the big context, the big picture of things, which I think is important to give people a context in which they do their work. I think if I were to do it again, I would probably want to create even more freedom and flexibility for people to do things their own way. I think that one of the things that the web has taught us is there is tremendous power in people collaborating without top down leadership. I think a lot more can be done in companies to create an environment where the leader can get even more out of the way. Ed Lazowska, he was the former chair at the Computer Science Department, University of Washington, recently used this analogy which I think is great. He said, "Running a research organization is kind of like running a cemetery: you are over lots of people and nobody listens." [laughs] I think that if you have a bunch of really smart good people, give them room to be creative and do their own thing. I don't want to do it over again, but if I were to I would want to experiment with offering a lot more of that kind of freedom. People are always concerned about, yes, things can go wrong if you don't manage it more tightly. I think if you create the right kind of structures and the right kind of support, people can do a lot more, and finding a way to give more room is something I'd love to see happen.

HAIGH: All right. So I will finish up with the same two questions that I have asked all the interviewees. The first one is some extent following from what you were just saying, but maybe puts it in a broader context. Looking back at the whole of your career, what do you think your biggest regret would be, either in terms of something that you did or didn't do or perhaps just in terms of how something turned out that might have been beyond your control?

ERISMAN: That's a fairly hard question for me only from this point of view. I don't tend to spend too much time thinking about do-overs. Because if I thought a little bit, what if I had gone to Union Oil instead of Boeing, what would my career have been? It would have been really different. What if I had taken the position at Carnegie Mellon in 1978 and left Boeing at that time, my career would have been really different. So there were a lot of branch points that I hit where I made a choice and I'm sure it made a huge difference. But I don't have any regrets. I loved the job I had at Boeing, I love what I'm doing now. If I had not gone into management and focused more of my time in the sparse matrix world, that would have been fun. Maybe I would have less gray hair because of that because it's the people issues that are so hard. But I don't really have a regret in that sense. I think there were times on projects where I probably wanted to go my own way and didn't listen well enough, maybe that's a regret. It fits with what I was just saying of listening more. I think that's about what I would say.

HAIGH: So then the reverse of that then, over the course of your whole career, what do you think the single accomplishment would be that you are most proud of?

ERISMAN: From a technology career point of view, leaving away my family and other kinds of things which are very important to me, but from a technical career point of view I would say there are a couple things. One of them is the sparse matrix book, which in the end when it was done I felt like I had said what I wanted to say, and I was pleased that I had the opportunity to do that. So that was one. I would say the other one would be that I think I had the opportunity to create a really great organization within the Boeing company in technology that respected and valued mathematics, and that respected and valued people in mathematics. I really gave a lot of who I was to making that happen and preserving it and I was pleased with being able to do that.

HAIGH: Well, that concludes the questions that I had prepared. If you have any other thoughts or observations, now would be the time to make them.

ERISMAN: I guess the one thing that we didn't touch on that I think is important is that I have had some mentors along the way that have affected my life a great deal. I think of them a very diverse set of people. One of them certainly was my professor, Bob Lambert, at graduate school. Bob Lambert had worked in industry and he wasn't one that said to his graduate students that you are a failure if you don't go to a top name university. I know there are professors that are doing that to their graduate students today and I think it's a shame, because I think the idea of having a context for your work is so important, and I think I learned that from Bob Lambert. We had the opportunity in the mid '80s for four of his grad students to get together and organize a party for him at Iowa State, and it was a very, very gratifying event. He died before he retired, unfortunately, so we were really thankful that we had done this. And I still correspond with his widow so Bob Lambert certainly played a role for me.

A person in Seattle who had nothing to do with mathematics or technology was a fellow named Al Green. Al Green probably taught me more about life in terms of thinking about a whole context of life. He was a Christian mentor, and pushed me to think about every aspect of my life as a whole as opposed to thinking of it in little pockets. He is in his 90s today and still a friend, and he made a lot of difference. My wife Nancy has been an incredible support to me. She was willing to move to barracks housing at Iowa State University when our first son was just two weeks old so I could go to graduate school. She went with me with two sons and a two month old daughter to England for six weeks so I could work at Harwell with John Reid and Ian Duff. And she has been encouraging and supportive for more than forty years of marriage while I worked too many hours at Boeing, and then started new things post Boeing. I am very grateful.

But you look back on a career and you say, "Here I was majoring in math and minoring in journalism," and you say, "What is this?" Yet the writing I have done, the journalism training was incredibly valuable to me. The magazine that I put out now is based on the journalism training. The magazine that I put out now is based on my experience in technology which came out of mathematics. The work I am doing in business ethics came out of my whole life, but also my business life and also my technology experience. In some ways these threads are really interesting and fun. So I find that really enjoyable to think about.

One of the reasons that I am now teaching is that you can learn more by teaching than by taking a class. I joke with my students sometimes the reason I am teaching a class instead of taking it is because it costs less and I learn more. But the fact is there is a lot of learning to do. Some colleagues that I know actually retire, and I think that is a shame. Learning is really a part of what your life is all about. So I try to read a book a week on a wide variety of subjects from business to technology to ethics to Christian faith to novels to history. I think it's just a part of living. I think it's really a shame when I see a person who has retired. So anyway, I guess if I had any advice for someone I would say don't ever retire. All right, that's enough.

HAIGH: Well, thank you very much for taking part in the interview.

ERISMAN: It's been fun.

APPENDIX: Albert M. Erisman Resume

B.S., Mathematics, Northern Illinois University, 1962

M. S., Applied Mathematics, Iowa State University, 1967

Ph.D., Applied Mathematics, Iowa State University, 1969

Work Experience

April 2001--Present

Executive in Residence, School of Business and Economics, Seattle Pacific University

Primary duties are teaching one class per quarter in the area of business, technology, and business ethics. Engaged in faculty discussions on technology, business ethics, globalization, and the purpose of business. Led the Oxford Seminar (September 2002) for SBE students studying in England during the Fall term. Steering committee, Cineter for Integrity in Business.

Executive Director, Institute for Business, Technology, and Ethics (IBTE)

IBTE is a 501 (c) (3) non-profit corporation founded in 1998 to promote good business through appropriate technology and sound ethics. IBTE publishes *Ethix* magazine six times per year featuring conversations with business leaders, reviews, a forum, and columns on technology and ethics. The IBTE website (www.ethix.org) is accessed worldwide with more than 40,000 hits per month. Primary duties include writing, reviews, managing the books, the website, promotion and advertising, consulting, and speaking.

Consulting

Clients include TriWest, The TEC Group, Corillian, Lawrence Berkeley Labs, Deloitte and Touche, Boeing, and Hewlett Packard,. Also working with several startups.

January 1990-April 2001

Director of Technology, The Boeing Company

Created and managed a group of 250-300 advanced degree computer scientists, mathematicians, statisticians, and engineers doing R&D for Boeing in the areas of information technology and mathematics. Responsibilities included managing a budget of $45 million per year, delivering value to the company as measured by internal customers from around the company (greater than $500 million per year over the last three years), and creating research agendas to meet high priority business needs. Reported to the President of the Phantom Works (the Boeing R&D Center) over the final three years. A member of The Boeing Company CIO council. Selected as one of eleven inaugural Senior Technical Fellows of The Boeing Company in 1990.

Teaching

Adjunct teaching in the School of Business and Economics at Seattle Pacific University, and summer teaching at Regent College in Vancouver, BC.

Co-Founder, Institute for Business, Technology, and Ethics (IBTE)

Co-Founded IBTE with David Gill in April, 1998, and began the publication of *Ethix* in October 1998.

November 1984-December 1989

Director, Engineering and Scientific Services, Boeing Computer Services

Responsible for a staff of 150 advanced degree mathematicians and engineers supporting The Boeing Company and the commercial and government customers of Boeing Computer Services. Responsible for marketing technical services to external clients, including the BCS supercomputer service offering.

January 1976-October 1984

Manager, Mathematics and Modeling, Boeing Computer Services

Developed and managed a staff of 60-90 advanced degree mathematicians and statisticians providing applied research services to Boeing and the external clients of Boeing Computer Services.

Visiting Professor, Design Research Center, Carnegie Mellon University (Feb-June, 1978)

Writing, seminars, and teaching part of one class in numerical analysis while on sabbatical from Boeing.

June 1973--December 1975

Manager, Numerical Analysis, Boeing Computer Services

Managed a group of ten applied research mathematicians in numerical analysis. Started the international seminar series, and managed the company wide project in mathematical software. Adjunct Professor at University of Washington teaching numerical analysis.

January 1969--May 1973

Boeing principal scientist doing applied research in large-scale sparse matrix methods. Adjunct Professor at University of Washington teaching numerical analysis, and at Seattle University teaching electrical engineering.

Prior to January 1969

Graduate student, teaching most undergraduate applied mathematics courses at Iowa State University. Eighth grade teacher and coach (1962-1964) in the Kane County School District, Illinois.

Selected Publications

Books

Making IT Better: Expanding Information Technology Research to Meet Society's Needs; Computer Science and Telecommunications Board, National Research Council (NRC), 2000 (jointly with 15 other members of the Committee on Information Technology Research in a Competitive World).

Direct Methods for Sparse Matrices; Oxford University Press, 1986 (jointly with Iain S. Duff and John K. Reid).

Electric Power Problems: The Mathematical Challenge; Society for Industrial and Applied Mathematics, 1980 (edited jointly with K. W. Neves and M. H. Dwarakaneth).

Articles

"Looking to the Future of Aviation Management, Technical, and Professional Education: Using brain research, technology and marketing in the classroom of tomorrow," submitted for publication, with Peter Morton, Tim Brady, Jordan Le Bel, John Medina, and Patricia Henderson.

"Managing Corporate Knowledge," *Bridges: An Interdisciplinary Journal of Theology, Philosophy, History, and Science,* Fall/Winter 2001, pp. 201-222, with David W. Gill.

"Dissecting the New Economy," *Life at Work*, January/February 2001, pp. 40-47.

"Sparsity Structure and Gaussian Elimination, "*SIGNUM Newsletter, ACM,* **23**, 1988, pp. 2-8, with I. S. Duff, C.W. Gear, and J. K. Reid.

"Advanced Computing for Manufacturing," *Scientific American*, October 1987, pp. 162-169 (reprinted and updated as a part of the *Scientific American Trends in Computing* best computer articles, 1988, pp. 148-155), with K. W. Neves.

"A Structurally Stable Modification of Hellerman-Rarick's P4 Algorithm for Reordering Unsymmetric Sparse Matrices," *SIAM Journal on Numerical Analysis*, April 1985, pp. 369-385, with R. G. Grimes, J. G. Lewis, and W. G. Poole, Jr.

"Computational Kernals," *Vector and Parallel Processors in Computational Science* I. S. Duff and J. K. Reid, Editors), North Holland Press, 1984, pp. 149-158.

"The Boeing Mathematical Software Library," *Sources and Development of Mathematical Software* (W. R. Cowell, Editor), Prentice Hall 1984, pp. 321-345, with K. W. Neves and I. R. Philips.

"A Generalized Methodology for Modeling System Components in Power System Dynamics Simulation," *IEEE Transactions on Power Apparatus and Systems*, January 1982, pp. 136-146, with M. H. Dwarakanath, B. Dembart, K. Hemmaplardh, and J. W. Manke.

"Analysis of Descriptor Systems Using Numerical Algorithms", *IEEE Trans.Aut.Cont*., Feb. 1981, pp 139-147, with R. F. Sincovec, E. Yip, J. Manke.

"Sparse Matrix Problems in Electric Power Systems Analysis, " *Sparse Matrices and Their Uses* (Iain S. Duff, Editor) , Academic Press, 1981, pp. 31-56.

“The Use of Mathematical Software Outside the Mathematical Software Community: A Panel Discussion,” *Performance Evaluation of Numerical Software* (L. D. Fosdick, Editor, North-Holland Press, 1979, pp. 273-283, with B. Einarsson, S. Hitotumatu, L. Fl Shampine, J. H. Wilkinson, N. N. Yanenko.

"On George's Nested Dissection Algorithm," *SIAM Journal on Numerical Analysis,* **13**, 1976, pp. 686-695, with I. S. Duff and J. K. Reid.

"On Computing Certain Elements of the Inverse of a Sparse Matrix," *ACM Communications,* March 1975, pp. 177-178, with W. F. Tinney.

"Monitoring the Stability of the Triangular Factorization of a Sparse Matrix," *Numerische Mathematic* 1975, pp. 183-186, with J. K. Reid.

"Hybrid Sparse Matrix Methods," *IEEE Transactions on Circuit Theory,* November 1973, pp. 641-649, with Benjamin Dembart.

"Decomposition Methods Using Sparse Matrix Techniques with Application to Certain Electrical Network Problems," *Decomposition of Large-Scale Problems* (D. M Himmelblau, Editor), North-Holland Press, 1973, pp. 69-80.

"Exploiting Problem Characteristics in the Sparse Matrix Approach to Frequency Domain Analysis," *IEEE Transactions on Circuit Theory,* May, 1972, pp 260-264, with Gary Spies.

"Sparse Matrix Approach to the Frequency Domain Analysis of Linear, Passive Electrical Networks," *Sparse Matrices and Their Applications* (Donald Rose and Ralph Willoughby, Editors), Plenum Press, 1972, pp. 31-40.

Work in Progress

Books

Nine Good Reasons to run a Business Ethically

Seven Reasons Why Most Large-Scale IT Systems Fail

Direct Methods for Sparse Matrices, second edition (with John Reid and Iain Duff, Oxford Press)

Business Ethics (based on recent lectures in China, to be translated into Chinese)

Recent and Selected Professional Presentations

Presentations on business ethics, multiple universities and public forums, March 2005, Hong Kong.

“Recognizing Corporate Immune Systems,” Invited Keynote presentation, Society of Consulting Psychology Midwinter Conference, February 11-13, 2005, San Antonio, TX

“Why Be Ethical in Business,” Serial Lectures given by Renowned Figures at Home and Abroad, sponsored by the State Owned Assets Commission of the State Council, September 7, 2004, Beijing, China.

Presentations on business ethics, multiple universities, September 6, 8, 10, 11, 2004, Beijing, China

“Ten Principles of Ethical Business Professionals,” Workshop presented at the Ambition, Money, and Self Conference sponsored by Eagles Communications, July 17, 2004, Bangkok, Thailand.

“Finding Context for Technology and Ethics,” Seventh Annual Ethics and Technology Conference, Chicago, IL, June 2004, Invited Keynote Address.

“Analyzing and Responding to Different Christian Views of the Corporation,” presented at the “Faith, Profit & Decision Making” Conference, sponsored by Wheaton College’s Center for Applied Christian Ethics, March 2004 (jointly written with Denise Daniels, Randall Franz and Kenman Wong).

“Seven Success Factors for Large-Scale Information Systems,” presented at a Think Tank at the UBS Leadership Development Center, Switzerland, September 2003.

"The Technology Factor in Business Trust," Top Management Forum, sponsored by the Turku School of Economics and Business Administration, Turku, Finland, February 27, 2003.

"Ethics in Business: Business Benefit or the Right Thing," Workplace Integrity Conference, Regent College, Vancouver, BC, February 8, 2003.

"Staring Down the Barrel of a Cannon," Breathing Life in a Web Based Economy, sponsored by IVCF, Cambridge, MA, April 2000.

"Technology Transformation of the Workplace," Fifth Annual International Conference Promoting Business Ethics, sponsored by the Vincentian Universities in the United States, Chicago, IL, October 29-31, 1998.

* * * * * * * * * * * * * * * * * * *

“Computational Kernals,” Vector and Parallel Processing II, Oxford, England, August 28-31, 1984.

“BCS Mathematical software for the Cray-1,” Cray Symposium on Science, Engineering and the Cray-1, Minneapolis, Minnesota, April 5-7, 1982.

“Why Scientists Do Not Use Software Other People Write,” AAAS Annual Meeting, invited Talk, San Francisco, CA. January 1980

“The Use of Mathematical Software Outside the Mathematical Community,” Chairman of Panel Discussion, IFIP Working Group on Numerical Software, Baden, Austria, December 1978.

“A User Oriented Multi-Level Math Library,” Mathematical Software II, ACM-SIAM Conference, Purdue University, May 1974 (co-authored with E.G. Cate, P. Lu, and R. M Southall).

Selected Recent Professional Activities

Washington Technology Center board of directors, chair of the Technology Business Development committee, 1998-present.

National Institute for Standards and Technology review board, Chairman of the Information Technology review committee, sponsored by the National Academies, 1999-present.

Board member, Institute for Business, Technology, and Ethics, 1998-present

National Academy committees on supercomputing (early 1990s) and information technology in a competitive world (1996-99).

National Science Foundation committee on the state of mathematics (1994-98).

Review committee for the School of Computer Science, Carnegie Mellon University, 1990-95.