An Interview with

FRANCES E. HOLBERTON

OH 50

Conducted by James Ross

on

14 April 1983

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

Frances E. Holberton Interview
14 April 1983

Abstract

Holberton discusses her education from 1940 through the 1960s and her experiences in the computing field. These include work with the Eckert-Mauchly Computer Corporation, David Taylor Model Basin, and the National Bureau of Standards. She discusses her perceptions of cooperation and competition between members of these organizations and the difficulties she encountered as a woman. She recounts her work on ENIAC and LARC, her design of operating systems, and her applications programming.

FRANCES E. HOLBERTON INTERVIEW

DATE: April 14, 1983 INTERVIEWER: James Ross

LOCATION: Potomac, Maryland

ROSS: This is a tape-recorded interview with Betty Holberton in her home conducted by the Charles Babbage Institute and I'm Jim Ross. Betty, how did you get into information processing, and how well did your previous education prepare you for what you found?

HOLBERTON: Well, information processing was not even conceived of as a discipline actually at the time I got into it. Originally, my family were scientists and teachers--mathematics, physics, astronomy. I was fortunate enough to go to a very good prep school, George School, which is a coeducational Friends boarding school in Bucks County and got a good founding in essentially the thinking process. I received a scholarship in the University of Pennsylvania and looked for a job. However, when I was in college I was told to major in mathematics. They said, "You know, when you take your exams, mathematics is what you should take." So I started with mathematics and I took analytic geometry with a very well-known Russian who had no time for women. Every day when he came into class (and there were only women in class) he'd say, "You women should be home raising children." Well after you have four months of that every day, and you go and you find out the next semester in mathematics is taught by the same fellow with no alternatives, I just gave up and decided I'd get an education. The only way I knew of getting an education that would get through that system up there was to go into English and journalism. This would allow me to take something in every discipline in which they would expect you to know something. I didn't know what I wanted to do. I knew I didn't want to be a teacher. So, that is the background of how I got into majoring in journalism. Then I got a job with a farm journal magazine, thinking, "Well, I'm going to get in journalism--I'll get in the statistical department and do surveys and things like that and get into the writing end." But I got so intrigued with the equipment and setting up...this was Remington Rand Powers equipment in those days, this was 1940...

ROSS: Tabulating equipment.

HOLBERTON: ...tabulating equipment which was different from the IBM, a different philosophy. I saw so many things that were so bad in the human engineering aspect of the whole way in which the survey had been established, that (I was just paid 15 dollars a week) I went into my boss who was a professor at the University at...

ROSS: What's his name?

HOLBERTON: His name was Franklin Cawl, Dr. Franklin Cawl. In fact, I'd been very fortunate. I came across a list of all these nice PhD's that I've worked for all my life. It's rather impressing.

ROSS: What was the Russian's name, the mathematics professor?

HOLBERTON: Shohat, Dr. Shohat. I was just discussing him with a friend of mine. Shohat was of the old school--he spoke to his wife in front of her face as Mrs. Shohat and [was] very formal and bowing and all that kind of thing. But he thought women should not be in education and should not get an education. That just turned me off; I was one of eight children and had a very delightful home life--four boys and four girls--and my father always felt that education was terribly important. I noticed that through the whole genealogical study I'm doing that education was terribly important. My father would encourage us, so when I got this job at the Farm Journal Magazine I started working in the area of this survey. I found out that the girls were making terrible mistakes and the survey was coming out with numbers like the lowest income family would use ten lipsticks a year. I just thought that something was wrong about that and I went back and look at the punch cards. They were all off punch, punched in the wrong column.

ROSS: Did this--was this a bias in the equipment as well?

HOLBERTON: No, no bias in the equipment. It had to do with the fact that the keypunch operator had to position it into a certain field character position before she punched. Therefore she had to count over and tab over to line it up so it would be one instead of 10 or 100. This made a tremendous impression upon me; [it was corrected in] the

original Unityper so the girl would never have to do that. She'd type what she saw and hit one key and it would move it over automatically. So the Unityper 1 was built on that experience I had.

ROSS: See, I guess I would say that's both a human and an equipment problem.

HOLBERTON: Well, it's an equipment problem because the equipment wouldn't do what the human wanted it to do.

ROSS: Exactly.

HOLBERTON: Okay. That's the opposite way around. That's saying that the guy who's building the equipment understands what the problems of the user is going to be. And that's the whole thing.

ROSS: That's what we're going to get to.

HOLBERTON: Right.

ROSS: Okay. So you walked into your boss.

HOLBERTON: So I told him I wasn't going to support these figures unless they brought in a whole new set of equipment that would verify everything the girl punched. And by golly he bought it. We started over again, which was amazing. I just felt so strongly that it--I had to be able to substantiate the numbers and if I couldn't, I just wasn't going to do it. That's how I got into thinking about that kind of a problem.

We always had desk computers at home. Since my grandfather was an astronomer, he was always working in chemistry and that the chemical tables or whatever they are--the elements tables and calculations, and so we had two machines in our home. One was an Olivetti that had stylus on it and you cranked the wheel; that went to my uncle and I never did see that afterwards--he used it in law court. I thought that was just great he had a calculator in the

law court. We also had a Marchant, and my brother and I, he was 13, decided that the divide circuit wasn't working one day, and we took the whole thing apart and see if we could fix it. I think I must have been about 15. I labeled every part and diagrammed where it was going to be, we laid it all out on the table, we got it all back together again, but it still didn't work. There was something that wasn't there that we didn't know about, but I had a feel for that kind of thing. I think that is the way I think.

ROSS: Mechanically oriented.

HOLBERTON: It's taken me about 30 years to figure it out, but I think differently from most people. I think kind of like a radar screen. On any problem I have I go around and around the problem until I have all facets of it well in my mind before I either sit down to write anything out on how it's going to be solved. I've done that in all kinds of things; I've seen logical errors immediately where I have had to spend hours describing to other people what was wrong, so I know I have something a little different. One time I said to Dr. Mauchly, "You know I never had a course in formal logic. I think I better go take a course," and he wouldn't let me take a course in formal logic. "Well, if I can't take a course in the thing, so I'll go find a couple of books." So I never had a course in formal logic. I just think the problem through. I probably do in the most bizarre kind of way, but it's helped me so far.

ROSS: Can you give me some specific examples from your career with computers about being able to locate logical design errors.

HOLBERTON: Oh...

ROSS: Well, you were on the other side, this is a leading question.

HOLBERTON: No, well, I wasn't really on the other side because I knew every gate number down to the last eye ball; I mean I needed [to know] the thing inside and out.

ROSS: Well let me put it this way, that's exceptional.

HOLBERTON: It is exceptional.

ROSS: Because most of the people who were working in programming or in mathematics would have been separated from the logic designs.

HOLBERTON: Right. Well, many times I was separated from actually having done a problem and then had to be brought in to determine what was wrong. Oh, many, many times. In fact, it almost got to the point where if the programmer and the engineer couldn't resolve what the problem was, then I was brought in. I thought other people had to be trained to think, to follow through on a problem, rather than me stepping in and, within a short period of time, finding out what was wrong.

ROSS: Sure that would do no good...

HOLBERTON: In fact, I made it a point when I moved from Univac down to David Taylor Model Basin to give the maintenance engineers three or four hours to try to find out what was wrong before I would come in, because I felt they really had to learn.

ROSS: Give me some specific examples from the ENIAC or UNIVAC machines.

HOLBERTON: Oh, the ENIAC was probably the first chance I had to explore detail analysis as in determining where something really could be pinpointed as being an error. Betty Bartick, whom I worked very closely with, and I were very, very good at telling the, not engineers, these were Army -- A guy by the name of Goldstine (not Herman) and Homer Spence, who were the two fellows who maintained the machine. We got to the point where we could tell them what tube was out. In other words, before we moved that equipment down to Aberdeen, I made sure that Dr. Deitrich signed a paper that said every circuit was going to be checked out before we moved it, every wire, every Jones plug.

We did that so we would get rid of any of these cold-soldered joints and things like that because they had plagued us. I found it was like a puzzle, I mean I loved solving puzzles. In fact, when I hired people for Eckert and Mauchly the first thing I would ask them, "Did you like plane geometry? How well did you do? Did you like that kind of thing?" To me it was essentially doing a very routine type of thing but at least moving it one step further and throwing the false starts aside as you moved forward.

ROSS: Narrowing down.

HOLBERTON: Narrowing it down. And not jumping to conclusions. So many people jump to conclusions, I almost did that when I bought my new computer. That's where this word debugging comes from. Not debugging but breakpoint. We would narrow our problem down by pulling the circuit. In the modern day computers (they have been for the last 30 years) the termination of one instruction starts another. Well, we had to have wires to do that, and that's what we broke. That connection and started the next one and here where it's a whole problem was right there on all 20 accumulators showing us what the values were. And so we can narrow it down by running it at full speed and stopping at various points along the way until we actually saw what was happening. That's really the first chance I had at taking a large mass of equipment and determining at what point things went wrong.

ROSS: Do you have documents relating to this? Or is this something that would be difficult to document?

HOLBERTON: We weren't allowed to write anything. That was the thing--that's the reason I held so much stuff.

ROSS: At Eckert Mauchly you weren't?

HOLBERTON: No, it wasn't at Eckert Mauchly, it was at...

ROSS: At David Taylor?

HOLBERTON: No, no, down the ENIAC this is the ENIAC, Moore School this is all secret. And we were not allowed to have anything we were not allowed to keep any notes or anything at all. I don't know how I happen to have that one sheet of paper I don't know. But that impressed me so much, that all this time during the ENIAC patents which I was being called in for seven years in a row to try to remember you know what happened on what day was really exasperating, so I decided that I wouldn't throw anything away. And that's the reason I've got so much stuff. I did not write for publication because I wasn't out to push my name ever.

ROSS: Well, that's not the only reason to publish.

HOLBERTON: Well, the thing is that at a time when a company...you don't know whether what he has is classified because of patents pending. You don't know the status of that at any time. So you're really at a point where you can't hardly publish. In those early days it was kind of like that.

ROSS: I found that as a trend throughout industry. Yes, it's very common that that's the case.

HOLBERTON: But one of the things that I found that has sort of hurt things--the flow of ideas--is copyright. The copyright and patenting has really dried up the material for conferences that is really new. I worked very hard against copyright.

ROSS: On any of the early machines can you give me examples of logical design...

HOLBERTON: Flaws, oh yes.

ROSS: Yes, you're jumping the gun, go ahead.

HOLBERTON: No, I'll take the other one first.

ROSS: Well, I was going to ask you that.

HOLBERTON: Alright.

ROSS: Yes, it seems to me that the programmers and the designers were separate to a large degree.

HOLBERTON: Not at Eckert Mauchly.

ROSS: Oh they weren't.

HOLBERTON: No. I don't know exactly what department I was in. I never did know my title, but I worked directly with Dr. Mauchly and with Bob Shaw and Fraser Welsh; those three people I worked directly with. I had nobody in between. In fact, that's the thing that I think my father never could quite understand is how in the world I ever got this position. But you didn't major in that kind of thing. I said they have confidence in me. And so that was very close there and I think that is one of the good things about that equipment that we could bat ideas back and forth without a lot of formality. There were just a few of us doing that, and we worked together as a very close unit.

ROSS: That is exceptional.

HOLBERTON: Right.

ROSS: That was a characteristic of its size.

HOLBERTON: It was the size. The size, and the fact is that we all believed so much in what we were doing that we just worked together, that's all. I didn't know the first thing about electronics and I still to this day do not know electronics. I could follow a circuit but I couldn't tell you why it behaves the way it does other than from the logic.

ROSS: You joined Eckert Mauchly in...?

HOLBERTON: 1946.

ROSS: And you left to go to...?

HOLBERTON: 1953. I had been working in engineering, and they said they were going to move the programmers to Washington and I had been commuting from Washington for three years at my own expense. So I thought, "Well, I'll get into programming and move to Washington," where my husband and I were living. Instead they moved to New York, and so that's when I left. I came down here and went to David Taylor because I thought they needed help.

ROSS: What gave you that indication?

HOLBERTON: What, that they needed help? Well because they were all new people and they hadn't been in on the design and we hadn't had for three years communications with anybody there because they had just started. We had the Air Force who were working with and the Army Map Service who we were working with. We had been working for a long time with other people before the machine was finished, and this was number 6 and they were being delivered and they were just trained and there they were. So I felt this gave me a chance to do that.

ROSS: Before you left, what other things changed with the Remington Rand acquisition?

HOLBERTON: Well, the whole thing changed. It changed actually when this (I don't know how much you know of the history of Univac) fellow from Totalizator, Strauss, who was a very marvelous person...

ROSS: What was his first name, do you remember?

HOLBERTON: No, I don't, I don't remember his first name. But he would come in maybe once a month and he knew

all of us by name, he knew what we were doing, he was interested in what we were doing, and he was the most marvelous person to work for because you felt that he was interested in what you were doing. But when Univac bought it, or Remington Rand bought it, things changed completely. Absolutely. The sales people got in, and the whole thing just changed. Women, as far as I could see, had absolutely no future under Remington Rand, absolutely none. I would go out on various things to try to sell computers and what not and the salesmen would pretty soon know that I was the only woman salary coming out of Buffalo which meant you were a certain grade above the secretaries. This was so annoying that I could see the writing on the wall; I really could. I knew that since I didn't have a Ph.D. and I had nothing to go on, that it just wasn't going to work.

ROSS: What of this has to do with Remington Rand's inexperience in dealing with this kind of machine or with that kind of marketing?

HOLBERTON: No, I think the thing was it was more their whole philosophy of sales, and just bumbling. Here they had what we thought was a far superior computer than the IBM 701 from the whole design philosophy and everything else and yet the 701 was selling and Univac was not, and this was annoying to all of us. It was just their approach to how they did it. They were not good salesmen. We kept telling them they ought to go to IBM school and be a better salesman. So it was really their sales force I think that made a lot of us unhappy with the situation. In fact I've come across some notes that went back and forth between us at the time and how we were criticizing them. It took them years to really understand and they were bumbling for many of the years, although they had a really solid foundation in the electronics area, they really did. In fact, it was many, many years before I ever heard of a new idea that wasn't discussed back in '46 or '47 with Eckert and Mauchly. It was amazing the various kinds of things that were just too expensive to do that way and all kinds of things. I mean even micro-programming, we had a design going on simultaneously with the Univac I that was basically micro-programmed. We didn't know how we were going to feed this thing in, but the whole logic was set out in the micro programming by Betty Bartick again and Art Goering, and so here we were, really far ahead in ideas because people were just sprouting them all over the place. It was a bright group of people, it really was.

ROSS: The people I've interviewed from ERA had the very same feeling, that they were unable to communicate the full potential that they had...

HOLBERTON: That's right.

ROSS: And therefore it was a frustrating situation.

HOLBERTON: In fact when I went to work at the Bureau of Standards I was able to read some of the documents, since the Bureau of Standards was the one who somehow had deals with...

ROSS: Issued the contracts.

HOLBERTON: That's right. I read those things and I learned things that I didn't know about at ERA and also at Univac at the time because it didn't filter through to me.

ROSS: Such as?

HOLBERTON: Well I think--I don't know if I read that there, but I always had the feeling that they lost the Nielson contract for Univac on the basis of work that I had done out there in 1950. I spent quite a lot of time out there analyzing their survey for purchasing, house purchasing and radio, these little things that hooked on radios, and TV's were just coming in then and this sampling kind of thing. I improved their system about 15% by using cards and I always thought that was the reason they didn't get the contract, but I found out that wasn't it at all when I read Nancy Stern's book. I only knew one side of the story. So there were lots of things that had to do with management that I had no dealings with at all.

ROSS: What about your colleagues? It sounds as if they were pretty separate from the management as well.

HOLBERTON: Oh, yes.

ROSS: How accessible was management? I mean you weren't very far away.

HOLBERTON: No.

ROSS: Norwalk.

HOLBERTON: Well, Norwalk, yes. But the thing was we were just peons, we were workers. When Univac came in, they set up this hierarchy, and I finally had a boss that I worked for. That was a disaster, too, because I'd go to him with decisions and he'd make the wrong decisions and I had to live with them. So that was an experience, too, which I hadn't been through before. The whole echelon of Norwalk, I had absolutely no contacts with at all.

ROSS: Did you ever go to corporate headquarters in New York?

HOLBERTON: Never, never. The nearest I got would be when I went to Springfield, Illinois, where I was trying to put in the first commercial Univac for Franklin Life Insurance. I had to go and visit the salesman everyday when I went out there, and I went out there every other week, flying on United and then Ozark. I would have to stop in on the salesman, and the only thing he wanted to know was when was he going to get his commission, every time. So there was no feeling about anything except what they were going to get out of it.

ROSS: And when you say salesmen, these were tabulating salesmen?

HOLBERTON: Tabulating salesmen, exactly.

ROSS: Okay, there's your problem.

HOLBERTON: There wasn't anything else except tabulating salesmen. No, there was no organization to sell this thing. In fact, selling it was a very difficult thing, it really was. As a data processor, it was difficult; as essentially something for the scientific people, no. They believed in it, and that we could see all the time, but they couldn't justify the money. I remember going to Martin Marietta here in Maryland with Dr. Mauchly one time, and the engineers were fabulous, and all we were talking about were paper products at that time. They didn't have the money, but they had the use for it. The people in the accounting area couldn't see it, they couldn't believe that you could ever even write stuff on magnetic tape, that to me was that's...

ROSS: Is it in the Tropp interview that you demonstrated that to someone once?

HOLBERTON: Yes, somewhere in my basement, I've got to find this, I have a magnetic tape in the basement, one of the old heavy reels.

ROSS: Yes, the clunkers.

HOLBERTON: Clunkers, yes, that's right, they were. We recorded on that, on this servo-mechanism that Fraser Welch had built (this was before we had the Univac) and we put on iron filings that picked it off onto scotch tape and then sandwiched another piece. I had that thing rolled up with a rubber band. I would take that, a magnifying glass, and the pulse code to the equipment whenever I went to show people that it really was there, and that the value really was in those bits.

ROSS: Did they understand?

HOLBERTON: Well, they could see it, I mean, at least it took some of the mystery away from it. But as I recall they were more afraid that you would lose material that went out on this thing, and also whether or not it was legal.

ROSS: Really?

HOLBERTON: The legal problem was fantastic, yes it was.

ROSS: That's astounding, I never knew that.

HOLBERTON: The insurance company would say, "Well, you've got to have it legal; you've got to have a document. You can't have all this information stored on something that nobody can see." There the legal problem would come in. So between people not believing and the legal department coming in, we really were sunk.

ROSS: And yet the insurance companies were some of the biggest...

HOLBERTON: They were, but that was three years to try to get them connected to it, that's right.

ROSS: You said lots of these places didn't have the money. The government did.

HOLBERTON: Yes.

ROSS: It got it one way or another.

HOLBERTON: That's right.

ROSS: And that's a tale in itself.

HOLBERTON: Well, the Census contract, of course, put us on the map there.

ROSS: What was it like working at the NBS?

HOLBERTON: NBS?

ROSS: You said you went to the National Bureau of Standards.

HOLBERTON: In '66; David Taylor first.

ROSS: Oh, I'm sorry. Okay, and then they were on a contract with...?

HOLBERTON: No, David Taylor had bought machine number 6.

ROSS: That's right. That was installed in the year you went.

HOLBERTON: 1953, so I went down there with that. My husband was already there and so I went there then. What was it like?

ROSS: Yes.

HOLBERTON: Well, I had an ulcer in the first year. It was very trying. I guess it was a matter of growing up and getting into an environment that I hadn't been in before; that is, competition rather than cooperation. When I got there we immediately got a crash program to be done for the joint services, it was an insurance problem. They wanted to do all these tables for the service people--annuities and things like that.

ROSS: Pensions.

HOLBERTON: That kind of thing, right, and it was a project that was dumped on us. Eisenhower was supposed to appoint the committee for the three actuaries to choose the tables. He didn't choose them for months on end and the due date was something like October 19, something or another. One month before the thing was coming on we didn't

even have the tables to deal with. Fortunately I had been on this insurance problem out there at Franklin Life for a couple of years. I didn't know anything about insurance when I went there, but I bought two books and I read those things solidly so I at least knew the terminology and knew what we were dealing with and that kind of thing; so I knew enough about the insurance business...

ROSS: And the kind of mathematical manipulations...

HOLBERTON: ...that's right. So I was asked to set this thing up as a program and get the whole thing [running]; a brand new installation, brand new programmers, brand new maintenance engineers, and a problem we hadn't dealt with before. So I did that, and it was really a very difficult thing with all the newness of the organization and everything else, but it really was done well and I got a commendation for it. Although the others got a group commendation, I was singled out.

ROSS: How did it prove the machine?

HOLBERTON: How did it prove the machine? It didn't just prove the machine, it proved me and my theories about how you take care of desk checking, flow charting, rerun procedures, the whole philosophy of tape labeling--everything that I had been teaching people you ought to do, I did, and it worked. It was [demonstrated] that if I hadn't done it we would have fouled it up any number of times, and so essentially I was reliving all the things I had thought about. It worked, but it worked to the point (since I got the commendation) that it made relationships a little bit strained.

ROSS: Oh, it did?

HOLBERTON: Oh yes.

ROSS: I thought it was going to go the other way.

HOLBERTON: No. Everybody worked at their competence at the time, whether how big a step it was for them or not. I mean it might have been a small thing but they did their best, and I recognized that. So, that is where jealousy comes in. And it was that actually. Although I stayed with it. I almost left, I almost went to C & O Railroad and Dupont. I have come across letters that...you know, but I didn't.

ROSS: Do you have documentation for this period?

HOLBERTON: Yes.

ROSS: So someone could go back in and discover your design theories?

HOLBERTON: Yes. In fact that's written up in a paper, a joint paper by two of the actuaries and it appears in a actuary journal and I have reprints of that. In fact, the flow charts are there. The amazing thing about that problem was that I did everything I felt had to be done, including document, even though it was going to be a one shot problem. Five years later I was asked to rerun that problem with a new rate table and the only thing I asked Dr. Policheck (who was marvelous as head of the laboratory), "Could I run it to verify that what's on the tape are all the corrections and everything as it came out the last time?" And it did. I hadn't made a mistake and forgotten something, and so it was.

ROSS: That's intriguing. You said that you were uncomfortable in this competitive situation.

HOLBERTON: Yes.

ROSS: That's often the case in industry.

HOLBERTON: I'd rather work with somebody.

ROSS: What was the organizational hierarchy at the David Taylor...?

HOLBERTON: Oh, there was a Dr. Policheck was in charge of the laboratory. Dr. John Wrench was my boss in the mathematics area, and I was supervisor of what they call research programming.

TAPE 1/SIDE 2

ROSS: I assumed that you might have had freedoms at David Taylor that you wouldn't have had in a corporate or industrial setting.

HOLBERTON: Well, I think because I was in an organization that had just set up Eckert and Mauchly, they were new, everybody was working together and what not, there wasn't this corporate thing. In fact, I can't remember anybody really telling me what to do. I just did what I felt had to be done and then went to somebody with what I had done. Because of the conversations we had back and forth, I would just go off and look at something or another. This followed through at David Taylor, also. I think I was very fortunate that I didn't have somebody really telling me what to do.

ROSS: Okay, that's what I was hoping to see.

HOLBERTON: Yes, unlike most people, that was one of the things that they let me do, and I thought that was important.

ROSS: Who assigned tasks? I mean, you said you went off and followed your own...

HOLBERTON: I was brought in originally as a consultant in the theory division, I consulted with anybody who had any problems about the UNIVAC, either programming or engineering or whatever. I started, you know, that way in

just sort of consulting. I remember the first six months I stayed around the computer to see how people reacted to it, when they were debugging at the machine because I had designed that console. In fact, when the machine came they asked would I just run the console for a week while this other fellow got up to speed; he had his course six months before and he had forgotten everything. Well, unfortunately for me, I thought of the console from behind the wires forward, and it was the most trying experience I ever had for that week because everything was a mental process that I had to figure out from behind and then look at it from the front and that was very difficult. But I watched these people and saw what they had to do in correcting programs, and in debugging, and that kind of thing. I got to work setting up a set of small programs, which were like the utility programs of today--the editing, all those things. I documented them and I picked things from all over the place, corrected them for the human factors aspect to make it easy for people to use and distribute that thing to the forty-seven UNIVACs around the world, and, in fact, I have a copy of that down there [in my basement], and it became the general utility system for the UNIVAC I...just by seeing what people were doing with the machine. I always was worried; did I put that switch in the wrong place, was it the wrong kind of switch?...this kind of thing. I always felt I made one major error in positioning of a switch. I put the general clear switch way up so the fellow would have to stand up and be aware that he was standing up to clear the total memory, and I used to say that he would do it so frequently that he would have to be jumping up and down all the time.

ROSS: I would have been more comfortable with it if I had to stand up.

HOLBERTON: Well, I thought you had to think about what you're doing if you had to stand up.

ROSS: Well, you've had a steady philosophy then about the user.

HOLBERTON: Oh, absolutely, from the very beginning. In fact that's the only reason I ever did that sort generator; I couldn't conceive of anybody having to go through the same thing that my group was going through in just examples of the same thing. It was sort of like a variation on a theme. I couldn't see anybody else having to do that when I knew it could be done mechanically by putting the pieces together with just the parameters that would come

into play with a different subroutine parts.

ROSS: Did it also have to do with the amount of storage as well?

HOLBERTON: No, we were so used to having a storage that really wasn't a problem. Our whole philosophy was we went from twenty accumulators on the ENIAC to a thousand memory locations in the UNIVAC. That was a big jump. No, we were aware of memory but we were able to design tape systems too; remember that the machine would read forward and backward and write forward, and that was one of the original philosophies that they came up with the original design, and that was so right. We learned how to manipulate that external memory very well. In fact, I think Don Knuth came across what he said was the first paper on backup storage that I had done back in '48.

ROSS: Other than trying to implement ease of access to the machine, what were you involved in at David Taylor?

HOLBERTON: Oh, at David Taylor? Well, let's see. I wrote an assembly routine with a mathematical library for the UNIVAC I which came with all the operating instructions written out for the user, which I had already done in the sort-generator. It always came out with whatever the thing was, because I felt that the documentation was going to be a problem.

ROSS: Grace Hopper makes that point over and over again in the Wexelblat book.

HOLBERTON: What?

ROSS: That documentation was essential from your side of things.

HOLBERTON: Yes.

ROSS: Go on with that.

HOLBERTON: That's the reason I went to the National Bureau of Standards. I didn't go there to work in FORTRAN at all. I had just spent three years correcting a computer sciences compiler, and I thought I had done an excellent job of organization and documentation. I felt that human communications was our basic problem. Whenever we went anywhere people couldn't understand what we were saying; the vocabulary was bizarre (it still is). We were always making up terms...very much a parochial kind of attitude in the computer field, and I think very much today.

ROSS: It does live on today; it's esoteric knowledge.

HOLBERTON: Yes. I felt that the people couldn't understand what we were doing because we were using such bizarre terminology. I felt that we would never get anywhere unless we could really make people understand, and maybe they could help us verbalize it better in the way in which they were thinking about it. So I felt that the human being was the thing to me that was so much a part of all my thought. This was true in even designing the UNIVAC C10 code. You would see the word "s" and it meant subtract, and "t" meant test, and this kind of thing; the choosing of the 26 letters of the alphabet so we would have a stimulus-response and you wouldn't have to figure out what in the world it was. That kind of thing I thought a lot about.

ROSS: You know there are some corporations today that are doing a little bit of that.

HOLBERTON: I think they're coming around. I think they are because they're now...

ROSS: Especially with word processing.

HOLBERTON: They're getting into the area where people are not professional programmers. They have not been trained, and they want the machine to behave the way they think it should behave. That's the reason I am rather intrigued by what a good job Radio Shack did with their Scriptsit. It's very difficult to get yourself fouled up and I'm rather intrigued with that. Somebody thought the system through. If you went around hitting all these buttons at

once you wouldn't destroy something, and you can't, and that's very good. [It is good to have designers spend a] little more time with watching people learn. When they get these small children now at a console they can get a lot of feedback there.

ROSS: You're frustrated though that it took 30 years.

HOLBERTON: I am frustrated, yes.

ROSS: Who were the kinds of people or the kinds of communities that you were trying to convince? Did you go before or communicate with computer scientists over the years?

HOLBERTON: No.

ROSS: In other words you thought it was a lost cause.

HOLBERTON: I thought it was a lost cause until people actually bumped up against it. I could see that you wouldn't get anywhere. In fact, I come from a family who were years ahead of their time. My grandfather talked about nuclear explosions on the sun in 1906 at the Philadelphia Philosophical Society and was laughed off the podium, and my father was the same way with team teaching at the Central High School, and here I was ahead of my time and I just didn't want to have to have the frustration of doing that.

ROSS: But there were others.

HOLBERTON: But I felt, you know, by just a few products that I could do in a lifetime might have some impact. [Interruption]

ROSS: I wanted you to tell me about the trouble people actually did have with some of the computers. You were

mentioning off-tape some horror stories.

HOLBERTON: Well one of the things that I do remember was that one organization did not have any internal labels on their information and they were depending upon these little labels you just put on the outside of the tapes. Things got scrambled so that they had to start over again, and this was a very large organization.

ROSS: And you wouldn't care to name it?

HOLBERTON: No, I would not care to name it.

ROSS: Oh, that's terrible. And now something like that is built into the system. You can't even store information on a micro without that.

HOLBERTON: That's right. In fact, when Dick Peterson who was with Univac left in about 1952, he asked for a set of my prescribed rules of how you should set up an organization, and I gave him all the stuff I had accumulated and one of them was tape labeling.

ROSS: How can you use random access well without it?

HOLBERTON: Okay. How can you do a lot of things without it? That's right.

ROSS: So what you're saying that the people absolutely missed the potential of the equipment that they were purchasing in some...

HOLBERTON: Well, remember random access was not even thought of at that time.

ROSS: Okay, place me in time here.

HOLBERTON: In that time everything was sequential, sequential files. In fact, it was essentially a replacement for the tab system when you get really get down in there, and our whole philosophy, that was what the marketing was--to replace the card. We had no equipment that actually could punch cards on the UNIVAC, it was completely key-to-tape, just like you would do in a computer now. It was all oriented toward replacement of the system without too much relearning the way in which they organized. The relearning would come after they realized that things could be done better. But you had to essentially permit them to take what they had done today and move it to a new piece of equipment and not change. Everyone requires a certain period of time for change; I don't care what it is. I think it's about a two year period that people will suddenly come and make the change and see [that] it's necessary. If you didn't have them change too much, and only change the equipment, you might get your foot in the door, and so that was kind of a philosophy there. But there had to be safeguards against human errors, both at the console and with this massive tape system and all these things that you were going to file away, with nothing visible. There just had to be some system for that, and this included rerun procedures, so that if the thing broke down you could go back and not have to start all over again. So I thought those things out very, very firmly because people would ask questions, when I would go out and sell this paper equipment. I would have to think through, "How would you do it?" The interaction with people alone taught me a lot about what to expect, as well as spending eight hours at night in bed pretending I was in one of these problems and had to get out of it. Just the thinking about the human problems had a lot to do with just everything that I did.

ROSS: Do you think that because so much of the demand for early machines came from very task oriented people, especially the government, that many of the ideas that you were thinking of weren't marketable at that time?

HOLBERTON: No, no.

ROSS: That was a very ambiguous question.

HOLBERTON: No, because the thing was, I was dealing not with the government's problems at all when I was at

Eckert and Mauchly. I was dealing with problems related to making an economic use of this machine in business, and the government was just a means of getting money. It had nothing to do with the kinds of problems they were doing, absolutely nothing. They would have been better off if they had a binary machine, in many cases, because the whole idea of how to handle tapes, how to organize a problem, this kind of thing, had to do with industry, and my experiences were with Franklin Life, Metropolitan Life, Prudential, and Nielson. They were the main people I talked with, and so the problems that they foresaw were things that I was dealing with; not government problems at all.

[Break in the interview]

HOLBERTON: ...because here the census--the 1950 census--in fact, I have all the programming down in the basement from that. Before, when they used to do it, they punched cards and all they did was to count, or sort and count, sort and count. Well the sorting was probably one of the worst things that the computer could do economically compared to a very cheap sorter, and we knew that. That was the reason I got [Tappet?] on that problem because I knew that that was going to be more expensive by machine than it was by [perhaps] going from card-to-tape and going out and putting it through cards and coming back again and sorting. That was one reason I worked in that area, just the sheer economics of it. In the case of the census problem, which was what I was talking about, the whole rethinking where the problem was done without a bit of sorting at all by breaking the addresses into two parts (which I had learned from the ENIAC--that is what Stan Franklen had done) and tabulating the information as if it were two different pieces of information in each storage cell. We were able to get it in there and accumulate the data. From then on I had a tape system which collected these things in parts and whatever it got collected at one time and that's as far as it would go, you'd hold it off until you collected all the parts of a state or an area or tract or whatever into one solid unit and then did the tables ? backward and forward through these tapes, so there's no sorting in it at all. Here was a complete change in looking at a problem, as against how we knew industry wasn't going to do that. This is one problem to be done differently. But we had to do that to just get the job done. They finally did two states on the machine in 1950. I remember Virginia was one of the states, I don't remember what the second one was, but they actually did use the equipment for that year. Then of course it's always been used ever since.

ROSS: People are resistant to change, in other words. It takes a while to...

HOLBERTON: I find this a very fascinating thing to deal with because I try to help them through the process of change, and this I've used in the last sixteen or seventeen years on the FORTRAN committee itself. That is, how do you approach people so they are not confronted with something that they don't want to deal with? How do you make it look like the idea came from them, and that I have been a past master at--make them think that they are the ones, by pulling them out, who have come up with a solution even though you knew the solution was there because you pointed the right questions to them. Yes.

ROSS: Exactly.

HOLBERTON: Yes. I found, being a woman, I had to deal that way. Yes. Because I knew in my heart, you know, when we came to the end of that FORTRAN thing what I worked hard on and got through. But I can remember one of the fellows from Honeywell saying...I was saying, "This is what I felt I did." He said, "Well, that's not the most important thing. The most important thing you did [was that] you kept us honest." And I thought that was the nicest compliment anybody could pay to me. I think the thing basically--sort of like psychology--is understanding people and how they think; because I don't think I think the same way.

ROSS: How different would your career have been had you taken another tact in terms of training. We started out by talking about your preparation, though you weren't attacking it as preparation for working in this. Say, had you gotten the electrical or mechanical engineering degree?

HOLBERTON: If I had gotten that I think it would probably not have been as good. We had a girl at Eckert Mauchly who had a degree in electrical engineering from Drexel and she was put in drafting. I think the fact that I was fortunate enough to be selected on the ENIAC and did a good job, and that Dr. Mauchly...although he wouldn't hire me--he wouldn't take me from Aberdeen--I was going up every weekend and spending all weekend with these

engineers up there discussing how to build this thing. Finally in February --I guess it was October to February--I said, "You know I really would like to work for that company." He said, "You would? I thought you'd never ask." I said, "I'm not in a position of having to ask somebody for a job. I thought somebody always hired you." He said, "I didn't want to steal people from Aberdeen," so that's how I got to work on it. I don't think it had anything to do with training, I really don't. I can't think of what more would have helped me in the kinds of approaches I had to take because I had to take it from the point of view of people and not of a dead circuit or a flip-flop or a relay. I don't know that the additional engineering from the point of view of what I could give, would have been any good. The other fellows were so sharp in engineering, I mean, I wouldn't have done a thing there. So I don't believe that was it at all; I think I was just fortunate to be at the right place at the right time and enthusiastic and that kind of thing. Because I lived it, I really did, before I was married; I lived it.

ROSS: What happened after David Taylor?

HOLBERTON: At David Taylor...

ROSS: You were there for how long?

HOLBERTON: Thirteen years.

ROSS: That's too long a period then to let go of so soon. Tell me some more about it.

HOLBERTON: About David Taylor? Oh, I haven't thought about that for years. Well, I spent, I guess, about two or three years on the LARC, this next machine both from the point of view of checking the design that was coming through from UNIVAC and doing software for operating systems. I've done just about everything in the field except a data base system. I've done an operating system, compilers, assemblers, and utilities and this kind of thing. I don't know when I got to the point of finishing that project (which was correcting this compiler; it was given to us because the LARC was late). Univac had a contract with Computer Sciences [Corporation] to build a FORTRAN and we had a

joint agreement with Livermore that we would come up with the same specs and they delivered the compiler, which was beautifully designed, very poorly documented, and patched at the machine level so we couldn't recompile it, or reassemble it, actually. We got a contractor in there with four extra people and my people and other people; there were seventeen of us who were to get that compiler to work. I spent about four months analyzing what the problems were and how to approach it without somebody goofing up because there were ten passes to it and a library. So I designed how we would approach them, how we would document it, and we had one person whose job it was to keep track that everybody did what they were supposed to do. It's a system that I have always felt very badly about not writing a paper because I think I could have helped somebody somewhere in looking at program maintenance. We had nothing; we didn't have all these intermediate stages of corrections. We had a base document which was corrected to the point where we could compile or assemble it after we took out the machine patches and we used that as everything, as our base. In fact, it was called standard for each base, that and the current compilation that we reapplied every time all corrections that worked to that one phase. They were stack cards that got continuously updated against the single one base. Anytime that we took out some code, we didn't take the code out, we made comments out of it, we left everything there. We introduced comments in the beginning saying what error we were correcting and then pointed to where it was corrected. We had trail box even on (?) the phone calls and things like that to be sure that we didn't in any way foul up. And that was another one that I thought I did a good job.

ROSS: And you said you documented that.

HOLBERTON: Yes.

ROSS: Well, good.

HOLBERTON: But I didn't write a paper on it. I didn't write a paper on it, and I've always felt bad about that. I thought it could have helped somebody but now people don't deal with cards. I think of all these corrections that go into computers and people saying, "Let's try this kind of thing." That's something I never did. I never said, "Let's try this." If I didn't know it wasn't going to work before I put it in, I didn't put it in. I would analyze it down and then

know whether it would work or not.

ROSS: Seeing that they don't deal with cards anymore doesn't totally wipe out what you're saying.

HOLBERTON: No, it doesn't and I was just thinking that in relation to some of these routines that update things. If it always updated against the same base, and you could always work for the same base and apply all the updates to that base, then you would be better off than trying to figure out when did you go wrong and have thrown away some listing somewhere. Yes. I thought about that the other day, because I was thinking about things that I hadn't done that I should have done. How could this be applied today? Because I do know that people have listings outside their doors all the time from computers, and it seems to me that you shouldn't have all this kind of garbage around. But you should be able to set up a system where you always have a full trace. In fact, there are two things that I think that I've been really thrilled with seeing come through. One's color on the screen. I had my room at the Bureau of Standards papered with maps, all kinds of maps, color maps, because in a map they have the ability to have color, larger font, wider lines, and everything to show you the bigger picture in the smaller thing, and I felt that we never would get the flow charting to do anything at all if everything is going to be black and white and they all looked the same weight. The color was a thing that we really needed; I was very glad to see that come through. Also I would also like to be able to see that you can in fact debug (and I really mean debug in a real-time system on-line) by keeping track of everything that has happened and you can just ask for it in an information retrieval system--ask for the activity at a certain point in time. I'd like to see that come through because programming in a real-time environment, debugging is miserable; it really is.

ROSS: Do you think there are any tools programmers used that are, any more tools you've already given me some, that are applicable today as well that are not being used?

HOLBERTON: I don't know; I think there's one tool that took at least twenty years to come forward. I was really annoyed at that because this was a tool that was done at Univac by a fellow by the name of Steven Wright, who was a very good fellow and he did a lot in the area of utility programs and this kind of thing. He wrote this program that

was an edited code and also gave all the cross references with instruction that was utilized. It was called an analyzer. It took twenty years before any listing that I ever saw came out of any system with a cross reference, and I thought that was an awful long time because I couldn't have lived without that piece of software. If you typed something and it made a mistake it was obvious it was referencing something in the middle of something, and it was so good a utility that we never even went on to the machine at David Taylor until we had checked our analyzer listing against what we wanted the program to do. We didn't go on and try, you know. We did desk checking and we did go through this process before we went on a machine so that most of the major clerical kinds of things were picked up. It was just the really logical ones that we had to deal with. That was really a philosophy on how to approach programming.

ROSS: Yes it does sound exactly...

HOLBERTON: I know that when I went had this original job for Eisenhower, essentially, I had all kinds of (in fact I think I still have in the basement) people's names against who programmed what, who checked what, who documented, this and this kind of thing, so that I can make sure that nothing was flubbed up along the way; because I was scared, I really was.

ROSS: Oh, an awful task. What influence do you think hardware has had on programming or conversely programming's had on hardware?

HOLBERTON: I think it's more the reverse. I think it's more understanding what programming is that has affected hardware.

ROSS: I actually have it written down that way.

HOLBERTON: You do? Yes, and I have an analysis that I did of a general flow of a business program, and this was done, I guess, in '46 or '47, which I took what I considered to be the concepts that you dealt with in files. Maneuvering files and putting them out and what you had to do to keep track of things and the frequency in which

you did this kind of a do-loop, if you know what FORTRAN is. That kind of thing. It went through the various programs that determine what percentage of the program was involved in this concept, to determine whether or not it was worth building that thing into a piece of hardware. And that's what you do today.

ROSS: Yes, I see exactly what you mean.

HOLBERTON: Yes, in fact after UNIVAC was built I had instructions with the engineer called a FEH statement which was really like an indirect addressed thing which would have improved just the utility kind of updating registers. I affected what the machine was because of what had to be done. In fact, I can remember Eckert didn't believe that we needed that many statements in a program for instructions in the machine. When I came to having the equals instruction he thought that was just a nicety, that you could always put two things and reverse the register and come out with the two tests. Well, I finally convinced Bob Shaw that we're going to do it so often that we certainly ought to have it. But it was after that got into the design that Eckert said, "No more instructions unless you can show that it doesn't cost more than 2% of the equipment and saves 5% of the time. We had to cut the stop statement in there because we'd use it once.

TAPE 2/SIDE 1

HOLBERTON: The way you used to compute the cost of equipment was to count the tubes. That's the reason it was so far off of its original numbers. They had the idea that you counted tubes and you put some kind of factor on tube count. But I think its taken a long time for engineering and the user to get down to the nitty gritty. It wasn't really until the microcomputer came along. That really was an important factor. When I went down to David Taylor I was free of any manufacturer so the manufacturers would come to me with a design. And so I reviewed many, many designs for engineers. In fact, I remember the 702. They only built six of them. They had so many things that were really bad, but the 705 was good.

ROSS: What were the differences there? Tell me in terms of...

HOLBERTON: I don't have any idea. But I do remember being sent out to what used to be Thompson Ramo Woolridge and I was sent out there from David Taylor to review a project they were doing, and they were designing a real-time system. I think there were about seven engineers who were presenting this stuff to me, and they had a major flaw in their interrupt system which I couldn't believe was there. I discussed it with them and you know it was a long time before I was able to make them understand that there really was a logical flaw. I often wondered about whatever happened to that equipment. Well, I was just reading in one of the Annals of Computing just recently they never did put the equipment out. But that same thing happened on the Univac LARC. It had a logical flaw in the interrupt system, and I would talk to the engineers about it and they would send me back another program saying, "This is how you program at it." All they were doing was getting themselves deeper and deeper into recursive action. Well I finally went to top engineer and still couldn't convince him. A very simple change of a cycle delay in one statement would have allowed us to use it in a motor programming sense, a very simple little thing. They had the same trouble, the same logical error, in the real-time interrupt on the UNIVAC III, which I understood got them fouled up in one of their contracts. They could no longer run the problem cause they'd always get in the same problem every time they got to a certain stage in the program. And so, presumably, many people cannot deal with this thing of having different levels with interrupts. I often wished I had stayed in engineering. It would have been a lot more fun, maybe. But I've had an interesting experience I think, although I don't think that the experience at the National Bureau of Standards was what I hoped it was going to be. It was not one of the major thing that I...

ROSS: You went there in '63?

HOLBERTON: '66.

ROSS: '66.

HOLBERTON: I went there at the time when the Brooks bill came out, because I felt that if we did get to have some kinds of standards so that people didn't have to change completely when they moved around, it should not be done

in the military which basically was the impetus behind COBOL. I was on that effort, that it should not be done with a military enforcement behind it. It should be done in a voluntary way from an organization that's recognized as a standards-setting organization. My grandfather was the one who first proposed the National Bureau of Standards.

ROSS: He was?

HOLBERTON: Yes he was. In 1884 he had a conference set up of international, what they called electricians, they were physicists, so he could deliver this paper on establishing the National Bureau of Standards, and it became part of the document that went in with the declaration of it. His name was Monroe Benjamin Snyder. So the Bureau of Standards has always been in my background, and I knew what it was and I wanted to wind up my career there. That's the reason I went, but to work in documentation, human communications. Even the Bureau couldn't understand. They couldn't understand the idea that maybe you needed a different kind of organization if you were approaching a problem from a different direction. There is a different kind of documentation if you are designing and writing a program as against the fellow who has to look it from the other way--there's something wrong with a program that data is wrong. He needs documentation which shows not the flow of the program, but the change on data and I couldn't convince anybody that you didn't want to set up one standard or have only way you force people to think when I knew debugging was a thing which everybody was dealing with. We weren't getting anywhere in any tools to help them understand. They would give me 10,000 dollars a year and pretty soon I decided that wasn't worth it.

ROSS: What was your position? Did you have a title?

HOLBERTON: Yes, I was supervisory mathematician, GS15 grade 9 when I left. It wasn't the place I thought it was going to be. I thought there would be a lot of cooperation. It's very much like a university and competition in writing papers and that kind of thing, and I found that environment very cold.

ROSS: Not just that but it's quite hierarchical.

HOLBERTON: Oh, yes, oh yes.

ROSS: And that might from your previous experiences...

HOLBERTON: Well, it's hierarchical, but I think the thing was that you got into the position where people were attempting to make decisions above you with much less knowledge about the problem than you had, and this is very disheartening. There was no way to convince them, and I find that's hard to deal with.

ROSS: You have told me that was the case on a general sense on giving me the big picture, what about specifics? This is a problem that really frustrated you.

HOLBERTON: Oh it did, well even with the sort-generator, the thing I did back in 1952, I felt I should ask someone's opinion. I had two different designs, one which was three wide which means it's actually going in at a power of 3 series against a power 2 series, and I felt that's the one I should do. They were all ready to go and either one just coding from what I had done. I asked Joe Harrison what he thought I should do, and he said, "Well the census people only ordered four tape drives; you better do the two." I knew it was wrong--before they got working very far they'd have ten, and they did. I never got to do the other one because I wasn't going to do it for free and I had left there and they wouldn't give me a contract, although I was getting everybody on my neck. Do it, do it, do it; I wasn't going to do it. I had other things to do. But that was a decision that was wrong, and there have been many like that they (?) I felt if I just had a chance to talk with them, but always they were so busy, and they didn't really want to show their ignorance too that's another thing. I don't know.

ROSS: Was there any opportunity to make changes when people stopped writing code and starting working with languages.

HOLBERTON: Make changes in what, in computers?

ROSS: Well, towards the direction that you were concerned with?

HOLBERTON: Well, in COBOL I did not believe the philosophy of having to write so much stuff there. It's, again, a shorthand way of doing it. I did believe that the most important thing was to be able to have comments in a program. In fact I said, "There's no assembly language worth a darn if I can't write a comment. As long as I can comment somewhere in the program that says what it is supposed to be doing as against how it is supposed to be doing it." That's what I felt was wrong in all the things we were doing. We were always telling the machine how to do something and there's no clue in the program that says what it's doing, so you can't really analyze from the point of view of understanding. I don't know how you can do that but I've often thought that we ought to have some means of interspersing all of our documentation in some way right directly in the program and then flip a button and have it take it all off and become your first draft. I've always felt that that's what I wanted was to be able to take things that I wrote down at the time when I was thinking about them right there where it is and then have it assemble it into where the document should be. Then I could take that as a first draft and then rewrite from it. It would have been a big help. But this isn't a variant you were trying to get at. You want to know what...

ROSS: At the time, of course, people just don't stop and start doing one thing but what were you doing...?

[Break in the interview]

HOLBERTON: COBOL, I felt, was very important because of its ability to describe data. That had not yet really been done, to describe data, and how would you be able to describe data, and that's the area in COBOL I worked on, on what they called the data division; this was new. I mean, if they only use the FORTRAN statements in COBOL it wouldn't be that bad. They could have a do-loop instead of perform or whatever, that someone had already done it and showed it could be done. But as far as actually being able to describe a stream of information, I hadn't seen that yet. I was interested in working in that area. I felt that in that area if someone's going to deal with a problem, it's going to be a professional programmer. He doesn't want to write a whole lot of stuff, he wants to be able to visualize

the total concept within one focal point on a page, not go like this and move his head back and forth. That's the reason I felt the picture clause in COBOL was very important and I had to fight hard to hold onto that because Jean Sammet did not want it. But I felt if anybody who's looking at a program can keep his eyes focused in a narrow things and move down the page, that you would be able to grasp much more information than if you had to spend the time turning your head from left to right. If you do something time and time again, you sure don't want to do it the long way; that's what people would be doing even though you would like to have some means of translating your shorthand notation for someone else which is what I did when I was on the COBOL committee. Every time they put a new statement in I had my little old book here with a mnemonic code--see, I write this way and I know exactly how this is going to be extracted and to expand it in a routine. I would write to show that it was COBOL, although it wasn't written in COBOL because I wasn't going to write that much. I guess I worried a lot about acceptance of work from the user point of view. What's the good of expending a lot of effort if people don't want it or don't like it. If it isn't what they want, then you've been figuring the wrong problem, or the wrong solution to the right problem. Well I'm glad to see that they actually are considering the human being. That to me is most [important]. I felt very good leaving because now it's in good hands.

END OF INTERVIEW