INTERVIEWEE: Owen Mock

INTERVIEWER: Robina Mapstone

DATE: 12 April 1973

RM: Okay, let's get going in a somewhat chronological fashion. I guess the Army was the first place you were involved with tab equipment?

OM: Yes. It was in December 1944. I was thrown out of the infantry on account of my hearing, and because I was thrown out of the infantry, I was sent to a replacement depot. The only thing they knew about me was that I was able to type. One day they took everybody that could type and sent them to registration. Next day, the people that could type were sent to a machine records unit. I missed the prep for the registration and ended up in the machine record unit. I went there as a key punch operator. It was one of these mobile machine record units the U.S. Army had in France.

RM: Oh yes, those were the things that IBM supplied to the military.

OM: Yes.

RM: What kind of machines did you have in the unit?

OM: We had a 405, a 513 reproducer, interpreter sorters, key-punches.

RM: What were the calculations you were doing?

OM: What we were doing was personnel accounting.

RM: You kept tabs on people, where they were and ...

OM: Yes, that's right. Keeping track of the rank, amount of forces, there must have been about five or six of those machine record units in Europe at that time. We accounted for a hundred thousand troops, I guess. We did that during the war and after the war was over, we went through the places where they ship people back. One of the times I remember we were taking the trailers out and breaking up the unit. I was doing some sorting and one of the trailers drove off. I claimed ever since that I lost 2,000 men. [Laughter]

RM: I hope those 2,000 men were able to find you.

OM: They found them, although I was pretty upset at the time.

RM: Were you up front with those machines?

OM: No. The machines were actually about ten miles behind the lines when they first came in, they actually did get a little bit of fire, but that was before I came there. That was in Normandy. I was on the front line in the infantry. I keypunched the thing. I think I keypunched about a million cards. And I was trained as a tabulator operator.

After that, when I got out of the service, I went to Berkeley on the GI Bill and graduated in, I guess, 1949. The only job I could get was as a tabulator operator for Foster and Clyde.

RM: What was your degree?

OM: My degree was in math, and at that particular time it was hard to get a job.

RM: Was that because there were too many mathematicians or what was the problem?

OM: Everything was pretty bad right then, particularly in the San Francisco Bay area.

RM: If one could get a job, what kind of jobs were available for mathematicians?

OM: Teaching basically. There weren't too many scientific projects sponsored by the military, that hadn't really gotten started yet. Just a little bit of it was going on at the universities. Down here the Rand Corporation was doing a little, but even there the computing hadn't really gotten started.

RM: No, it was quite early. That is right. Had Professor Morton started on his machine? [Interruption] Were you familiar with Professor Morton who was in the Electrical Engineering Department at Berkeley at that time?

OM: Let's see. They were building a computer there.

RM: Yes. The CALDIC.

OM: Yes, I remember him. I worked with him. I did some free lance work in that lab.

RM: Oh, did you?

OM: Yes, because they had a 405 and later they had a 602A.

So I went to work for Foster and Clyde. I worked there for Foster about a year and a half at $200 a month. I did tabulating there. We worked with a 402 or 403, I've forgotten which. I believe it was a 403. Normally, on a 405, you could only take three sum steps really; minor totals, intermediate totals and major totals. On the 403, I think you had to have a 403 to do it, you could go into an interative sum, and just keep on summing as long as you wanted to. I worked out a board where we could tabulate by the month, get totals by the month, punch on the cards which month the billboard was supposed to be billed for, and it would go through all those twelve and print out a line for each one of the possible twelve totals on the card. That was my first program, I guess. [Laughter] That is my first thing that would really qualify as a program.

RM: We are talking about machines with plugboards, of course.

OM: Oh yes, yes. Removable plugboards have been around since the 1930s. Those 405s were pretty complicated. At that time I just did a little bit of wiring.

RM: Did you have a CPC?

OM: That comes later.

After about a year and a half, which brings us to about 1950, I went back to Berkeley to do some graduate work. I was there about eight months still on the GI Bill and doing free lance computing for other people on the Electrical Engineering Lab equipment. I was doing work for the Oceanography Lab there.

RM: What kind of calculations were they doing?

OM: One of the things I did was an auto-correlation analysis. They were using a 405 and a reproducer, using progressive digiting to do multiplication instead of just addition. I also did other expansion and polynomials on the 602. I had troubles with that one because the first board I wired I couldn't get in the machine; I had too many wires. So I had to simplify my part. [Laughter] You can do a lot of things with the 602A.

It turned out I was making more money between

the GI Bill, my free-lance computing, and going to college full-time than I was making when I was working at Foster and Clyde. So about September of that year, my wife got pregnant and I got kicked off the GI Bill because they said I skipped the summer semester. Dr. Lehmer was going down to the INA at the same time, so he offered me a chance to go down and work on numerical analysis. I started out working on the original EAM equipment and then I worked on the 602A and on the 604s, and I worked on the CPCs when they came.

RM: What type of work was the Institute doing at this point?

OM: They were doing scientific computations. I worked on one which was an analysis of a ... for Stanford Institute. The guy that designed it obviously didn't do any of the computing on it, because the expansion was just not the right way to expand it. Due to the usual lack of communications between the guy that is doing the job and the guy that designed it, I struggled with it for three to six months. We used all sorts of tricks and at one point I ended up with thirty-two significant digits of results spread over ninety-seven answers. [Laughter] That one didn't work out too well.

At that time Huskey was Director and Lehmer was Assistant Director. See Thompson was there. I didn't work with these people who were all associated with the SWAC. I had very little association with SWAC. My only association was with the assemblers and loaders. They took a collator and attached it to the SWAC and that collator was capable of reading binary cards. At first everybody punched the binary cards by hand which could be quite a task. Basically there are about 800 punches in the card. Normally, if you don't get them right, you have to punch the whole card all over again. However, we quickly learned how to take these little chips and push them into the card. They will stay at least long enough to reproduce the card and get a new one, although some people just leave them in there which is kind of dangerous, because every once in a while one falls out and you don't notice.

RM: And then figuring out where the heck it came from.

OM: Finally we took the 602A board and wired a plug-board that would convert from decimal to binary. We had a selector on this so that instead of punching a five the first card would punch a nine when it thought it was punching five, the second card thought it was punching eight, the third card thought it

was punching seven and so on down. This is the way we did it and as a result of that you would end up with twelve cards, one with the nines, one with eights and so on, each representing a conversion from decimal to binary. Then you would take the cards to the reproducers and gang punch the cards together. Then you ended up with one card which was the binary card.

RM: That's pretty clever.

OM: Well, that was. [Laughter]

RM: It sure beats sitting there and punching eight hundres holes in a card.

OM: Yes, it did.

RM: That is almost not feasible. It had more hole than card.

OM: Yes. That was a problem. We also took a 405 and had a very primitive relative loader.

RM: Oh, really? How does that one work out?

OM: I don't remember. That one is kind of vague. It was the most primitive type of a loader.

RM: Who were some of the customers who you were doing work for at the Institute?

OM: I had very little contact with the customers. The only one I remember was the Stanford Institute. I

did a little work on some tables for Grace Hopper, who was at the Institute at that time. Everett Yowell was basically my boss. He was an astronomer. Since then he has gone back to NCR. Fred Hollander was my immediate boss underneath Yowell, Fred is another astronomer. He was at UCLA the last I heard. John Postly was there at that time. Ed Osborne and I worked quite closely together.

RM: Basically you were doing scientific and mathematical work for government agencies?

OM: Basically, yes.

RM: Were you at this point in contact with the West Coast companies, Rand and the aerospace industries, and were you exchanging ideas and methods and plugboards?

OM: Yes. I wasn't directly involved with this at that time. I saw them come in occasionally.

RM: Really, you were down inside the machine playing games with it?

OM: Yes, yes.

RM: Were you called a programmer at that point?

OM: No, I was not. My title was a Mathematician.

RM: Had the term programmer been coined? Were people who put things into the machines programmers yet?

OM: No, I don't think so. Of course, the CPC is the

Card Programmed Calculator. They talked about program steps on the 604. I don't remember whether they called them that on the 602A or not. I can't remember. I don't believe I called myself a programmer at that time.

RM: [Laughter] Okay. On one of the tapes, I think you mentioned something about taking a programming course with Walter Cannon on the SEAC -- was that you?

OM: No. Cannon was there for awhile when I was there.

RM: At the Institute?

OM: Yes.

RM: Did you stay with the Institute till it --

OM: I stayed about a year and a half, until they folded.

RM: Do you remember the story about why it actually folded?

OM: It seemed to me they had another republican in office.

RM: Oh, that is a good reason! Dereck Lehmer can probably tell me that story.

OM: Yes. Economy, I presume.

RM: What was the next step in your profession?

OM: To North American Aircraft.

RM: What were you doing at North American?

OM: When I went there they were getting ready for the 701.

Charles Davis was in charge of their machine. North American was split and the computing equipment was in the accounting end of the business. In addition to the operators Charlie Davis wanted to get a small group of people who were in utility program and things like that. I went there, Don Brehind and Irwin Marshall were there, and basically there were the three of us. There were a couple of other people that I can't recall right now.

RM: The 701 was coming in, is that right?

OM: That's right.

RM: Had you at this point received the instructions?

OM: Oh, Ed Law also was there.

RM: Oh, yes. Had you received your Instruction Manual so that you were able to start working on the program?

OM: Yes.

RM: What was North American going to use the 701 for?

OM: For engineering.

RM: You were now going from a plugboard type machine to a true stored program machine.

OM: Yes. Actually we had one CPC that was wired to be a stored program computer.

RM: Can you tell me about that one?

OM: Yes. The CPC had what they called iceboxes, which were storage devices. We had, I don't remember, four of them maybe, and each of them held about twelve words. We could program so that you could sequence in from the icebox, pick up the instructions and interpret them. As you know, the way the CPC worked was to interpret the cards. We had wired it so that in addition to interpreting the cards you could have them call up instructions out of the iceboxes and interpret those as well as the cards.

RM: In other words the card would tell it to go to the icebox, and bring --

OM: Bring out some sequence of instructions. The instructions in the icebox could include loops and your cards could load into the icebox also.

RM: Okay, it could load information into what we now call memory.

OM: Right.

RM: The 701 was a ...

OM: Yes, that's a real stored program machine. It had 4096 half words.

RM: How did you go about preparing for this new development?

OM: Let's see, one of the people, Ed Law, wrote an assembler. This was a sequential, symbolic assembler.

The instructions had sequence numbers, not necessarily adjacent. The data areas were handled by relative binary. Of course we had to write a loader to go with it. We took the code from IBM, what do you call that?

RM: SPEEDCODE?

OM: SPEEDCODE, yes. We took that essentially as is. We also took some codes from Los Alamos. I think they called it Dual. Basically what we did was take the floating point code from Los Alamos and build a system of routines around it, using relocatable binary loader. We had standard print routine, standard dump routine, tape handling utility, standard card reading routine both binary, a routing that would read octal, a routine that would convert fixed field decimals. I guess we didn't have too much in the way of relocatable loaders yet.

These routines would load into fixed area memory which was about the first 2000 octal words. Let's see how much did we have. We had 4000 words of storage decimal. We had 4000 half words and we took about 512 half words for standard routines. We insisted that everybody use these routines, which is a little different from the way some of the other

companies operated. I think Douglas operated a little bit the same as we did insisting that people use these standard routines for everything.

RM: Why would some companies not have gone this way? What would have been the benefits of not doing it?

OM: More freedom! I guess one of the reasons why we developed this way was that operators rather than the people themselves were pushing the buttons. We had to have standardization and efficiency.

RM: So really you set up as a closed shop then?

OM: That's right.

RM: Was it always run as a closed shop?

OM: Yes, yes. We were the most closed of closed shops, you might say.

RM: In other words the engineer would come to the programmer, present his problem ...

OM: I'm sorry, we were a half closed shop. Yes, yes. The engineers wrote their own programs -- yes, we were open shop. That's right. But operationally we used operators only.

RM: Okay. So the engineer wrote his program but maybe consulted with you on ways to go and ...

OM: Very little of that because Frank Wagner's group in engineering was doing that. Quite a few people came out of Frank Wagner's group.

RM: Wagner was working with the engineers setting up the programs for their problems, right. Then you have the operating group that actually pushed buttons.

OM: Right, and then our function was writing utility programs for loaders, assemblers and so forth. Let's see, we also had a patch routine which would take a binary program, expand it, insert instructions right in it and relocate all of the information into a new binary program. It didn't work too well because people then found their assembly listings weren't reliable and that the locations weren't where they thought they were. To start with it wasn't bad.

RM: You mean you had to be very specific at that point about where you were going in memory?

OM: Oh, yes. Yes. Douglas had a little bit more relocation in theirs than we had, if I remember. It's been so long ago that it is hard to remember.

RM: The 701 which was appearing on the scene was something quite different to anything that has gone before.

OM: Yes, that's right.

RM: I'm trying to understand what were some of the problems you were faced with, and how did you go about them?

The tendency is to say that you wrote an assembler and you wrote a loader. You had to get to the point of understanding what it is you needed to do. I'm just curious if somehow we can recapture some of that by going back and trying to think, "Well, I walked in and here is this machine; now how am I going to use it?"

OM: We must have thought a lot about that. We were impressed by the speed of it. How do you keep it busy? That is why we did work so hard in the area of loaders and assemblers. That is why we ran the operationally closed shop. We worked very hard in getting fast turn around. We didn't want to waste the time of the machine. I had seen that on the SWAC, I'd seen what happens when people just stand around and push buttons. People response time is something awful. One of the things that I carried with me was the importance of being able to operate quickly. I guess I was interested in operating systems from the very beginning because of my first experience with the SWAC; the complete waste of the machine by the people sitting there and thinking. We weren't going in blind by any means because we had already had exposure. We were learning quite a

little bit on what had been done by the people at Los Alamos and what had been done in IBM. I guess Willard Verishious was involved.

RM: There is a name I don't know.

OM: He was first at Los Alamos and then with IBM. IBM had a sequential, symbolic assembly program. At that time we had some talk about simulating a CPC.

RM: Oh, that is interesting.

OM: In fact, I think we did have a program of simulating a CPC that was used for a while. I wasn't happy about that.

RM: Why would one want to simulate a CPC?

OM: Because it is easier to program, that is the theory behind it. People think it is easier to do it the old way.

RM: Did you get it running?

OM: As far as I know, I think we did use it for a while. Also, I think it may have had something to do with getting some of the CPCs out. I don't remember that too well.

RM: Somebody mentioned that you wrote the program on the 701 that would do arithmetic to 316 places. Do you recall that?

OM: Yes, yes.

RM: Maybe you could tell me a bit about it?

OM: It was a multi-precision program. What you did was to use words as pointers for the real information. The word would have the location for the real information, and it would have the number of words contained in the real information. Of course, it would have zero words, in which case it would just be a zero. It took a half word for each, I don't know what to call it, piece of information. The other part was just a question of writing the routines to do a multiple precision arithmetic: addition, subtraction, multiplications. Had no problem with addition, subtraction, multiplication, but I did have problems with division. Division by a single word was pretty straight forward, but I never did get division by multiple precision without bugs.

That's where I got into garbage collection. That's a term that they use. This type of thing that they ran into before in list handlers, as they call them. Quite a few programs operate so that you just leave the space laying around, you don't do anything with it until you run out of space. Then, when you run out of space, you have a thing that goes through and moves everything down in memory to retrieve

the free space. That's known as garbage collection.

RM: It is a nice term. That way you can totally utilize all your memory.

OM: Yes. Otherwise you just run out of memory.

RM: One thought that occurs to me is that what you had was a new toy.

OM: Yes, yes.

RM: It must have started a whole lot of ideas going in the minds of the people who were playing with this new toy of things you could do with it. I suspect some of them weren't all especially useful, but maybe humorous or just gymnastic type things.

OM: ... wrote an interpretive program and the way it worked is you wrote instructions, like assembly instructions, and then connected an amplifier to the computer and you could write music that way. Since that time it has gotten much more elaborate.

RM: What was some of the music?

OM: I don't even remember the music. You could write it very fast. Each line you wrote was a note. All you got was a simple square wave out of the amplifier. That was one of the earliest instances I know of where that was done. This brings us about up to the point of PACT-1.

RM: Was that the beginning of SHARE?

OM: No, PACT-1 was before SHARE.

RM: Okay, what was your involvement? How did PACT-1 come about?

OM: It came about because we felt that machine language programming was doing things the hard way. Actually we started just about the same time as FORTRAN did, maybe a month or so earlier or maybe a month or so later.

RM: Was it about 1954?

OM: We got our machine in July of 1953; maybe in the fall of 1954.

RM: Was PACT a joint project with other companies?

OM: Yes, yes it was.

RM: Who were the others?

OM: You can have this if you want it.

RM: Oh, great.

OM: This is the version of PACT that was for the 704. It has got lots of names on it.

RM: Thank you very much. It was a group venture to come up with a symbolic language.

OM: Yes, that is right. The thing that attracts people about a language, and it is still true, is the great big, long mathematical expressions you can write.

That is really not the hard part of computing; the hard part of computing is really manipulating the data. It is really fairly easy to write mathematical expressions. We ended up with a fixed formated language. It had a stronger indexing ability than FORTRAN, but unfortunately fixed format languages weren't in. They still aren't. So for that reason it never really caught on. Before that time an interpretive system had been written at MIT, and we knew about, that took mathematical formulas and interpreted them. We felt that really wasn't the hard part. The hard part was to index. They didn't have indexes then, and they didn't call them indexes. The 701 didn't have index registers; we were concerned about subscript manipulation. So this group of us developed a language.

Our model for developing the language was based on how would you go about telling the problem to the girl who used to have to do your computing for you. The question we kept asking is: How would you state a problem to her? We tried to model the language around that way of presenting the problem. We first programmed PACT-1 for the 701, and then we re-programmed it again for the 704. The PACT-1

compiler was actually working before FORTRAN, so this was the first true compiler.

RM: Was one of the problems that you didn't have the impact behind you of a whole corporation of IBM's size?

OM: That is right.

RM: Were there other reasons?

OM: I think that probably wasn't the main problem. The fact that it was fixed format meant it wasn't as "glossy," and I think that was the big issue.

RM: Would you just explain the term fixed format?

OM: Well, fixed fields. Like assembly language used to be.

RM: There is only one place for one piece of information, or one instruction.

OM: That is right.

RM: Where as in FORTRAN, you can ...

OM: You just write down a formula: A = x + z or so forth. In our case we had to take this formula and break it up, go through the individual steps and write down the individual steps to do the computations.

RM: So FORTRAN gave people more freedom.

OM: Yes. They probably made it a little more visible, to use today's terminology. It is easier to read

and understand. That is a little bit fallacious also, because it becomes very complicated. No language is easy to read and understand, but it is better than having a machine language.

END OF SIDE I

RM: We were talking about PACT-1. How did PACT-1 get distributed?

OM: How did it get distributed? We mailed them, directly.

RM: We, being the people in the companies who were involved in the development.

OM: Yes. One person collected the programs together and mailed them out. There would be a set of binary decks and we put the binary decks in the right order and you ended up with PACT-1. There were five phases if I remember rightly, and each one of the five phases was a binary deck. When you had some changes to your section, you mailed them to the guy that was in charge of distributing and he in turn would then mail it out to all the people.

RM: Did you end up writing manuals and instructions?

OM: Yes, we did write manuals, and we had to update as we went along. I don't remember whether we had any formal internal manual. There were two users manuals.

RM: Was this set up as any kind of financial venture?

OM: No.

RM: It was gratis? And it was only useable on the 701, or later on the 704, is that correct?

OM: That's right.

RM: In other words, it was a community venture for the

people who had the 701s. IBM didn't come in on this at all?

OM: That is true. We got a little moral support out of IBM. Actually we had some programming support from IBM on the 704 version from the local district office in Santa Monica. Has somebody else covered who was in the group?

RM: No. I was going to ask you if you recalled the names.

OM: Yes. Okay. It was divided into a Policy Committee and a Working Committee. Let's see if I can go through the Policy Committee. From North American it was Jack Strong, Frank Wagner; Douglas, El Segundo, Walt Spicer and Bill Debruski, John Lowe; Douglas, Paul Armer; Rand, Lee Amaya; Lockheed, California out in the valley; China Lake was involved but I can't remember who was on the Policy Committee. I think that is all. The Working Committee for PACT-1 was myself; a young kid [who] I don't remember (he's not young yet anymore) [Laughter]; Douglas, El Segundo, Chuck Baker; Douglas, Santa Monica, Irwin Greenwald; Rand Corporation, Roy Rigby from Lockheed, California; two people from China Lake.

RM: PACT-1 was for 701?

OM: That is right.

RM: What was IA?

OM: IA is the same language for the 704.

RM: Did PACT continue after the 704?

OM: No.

RM: Was PACT picked up by the SHARE organization as one of their programs?

OM: No. Not PACT-1 definitely. SHARE did not start until the 704. You'll find some other things as far as the relationship of PACT-1A to SHARE and to the IBM distrubution agency in these files. IBM said they would distribute it by request only. Conventionally SHARE stuff was distributed everywhere there was a member, but apparently IBM wouldn't do that.

RM: Okay, were there any other ventures of note through this period that we should be talking about?

OM: The 701 had the first operating system of the modern type; it was the modern type of operating system based on tape, what you might call a tape operating system. As I said, Douglas and North American had some utilities operating together as sort of a system already, but that was off cards. We got some of the new tapes that IBM came out with late in the

701. It was basically the same tape as the 727s, which are the things we have now. The earlier models had a different kind of a tape drive and it was not very reliable.

RM: It was the drive we are talking about, not the tape itself.

OM: No, the tape itself was different too. The whole arrangement was different. They also came out with peripheral equipment: off-line printers, off-line card readers, and that was the other thing which I think related to the tape operating systems. The fact that the computer was synchronous, and didn't have any really satisfactory bulk storage, dictated this class of operating system.

Our 701 operating system was designed to experiment with this technique because we discovered the problems that occur when you combine the computer to a 150 line a minute printer. So we developed an operating system that would go out from tape and come in from tape.

RM: That left the computing part of the machine free to do computing. Did IBM get into this themselves or were you really pioneering this idea?

OM: No. We were definitely pioneering, and we pushed IBM.

RM: Did you push them into coming up with some of the ideas themselves?

OM: Yes. That comes later.

RM: Are there any other 701 aspects?

OM: Nothing else that I can think of.

RM: The kind of problems that were being solved on the 701 were engineering, and the engineering was tied up with aircraft?

OM: Yes.

RM: Were there missile problems?

OM: No, not on the 701.

RM: Do you recall what planes were in production at this time?

OM: We were doing structures work, but I don't remember what the planes were. I remember one program, and was probably the program they got one of the parameters wrong, and according to the results the plane was flying backwards and one hundred feet under the ground. [Laughter]

RM: I guess there was a bug in that one. [Laughter] All right, what sort of happened then, the 704?

OM: The 704. The concept for SHARE really grew out of the PACT group because PACT had been so successful in terms of cooperation. Frank Wagner and Jack

Strong were largely responsible for starting the SHARE organization.

RM: Yes. History has fairly much captured that. The 704 comes into the picture and you were getting ready for a new machine. But earlier you mentioned that you had been responsible for pushing IBM into the direction of peripheral equipment and operating systems. Maybe you could enlarge.

OM: Yes, well peripheral equipment especially. I don't think the impetus for the operating systems idea came quite yet. At that time IBM was in the habit of assigning free computer time on their machine in New York to check out your programs before they gave you the machine. You have got two weeks, eighty hours or something like that of computer time. Usually they would give it to you in blocks. North American, and Rand (I don't remember how many), I can't remember if Douglas was involved, Lockheed decided that we wanted our time altogether and that we would keep track of the time between us. We got our time together and we worked out our own operating systems, I think we were working on PACT-1A at that time and that we did a checkout. We all got the machine together and we would just take turns. As fast as one guy would

get off, somebody else would get on. It was quite a contrast because at the same time that we were on, the people from -- what is that high security organization?

RM: Atomic Energy Commission?

OM: No. It is the one that is concerned with espionage and all that.

RM: Oh, the FBI?

OM: No.

RM: The CIA?

OM: Nope. It is the one before the CIA, Tom Steele used to be with them.

RM: You have got me.

OM: They had lots of money, and every time after they would get through they used to erase the drums and their tapes before they would get off the machine. They would come on for about four or eight hours, very leisurely, and then we would come on with people rushing in and out of the computer. [Laughter] We had developed and got working programs which wrote out the tape. After we had been on the machine about half an hour we would have a whole reel of tape to be printed. IBM [laughter] just wasn't equipped to handle this kind of an operation. This was one of

the first times they were exposed to what could be done about pushing work through by really having a system for doing it. At this point we were still checking out the system, but even when we were operating manually we were operating in this kind of a mode. A couple of us would act as operators and that is all we would do. A guy would come in and hand you the deck and we would run it through, go out through the tape, and do multiple jobs on tape outputs. We had troubles because we put stuff out on tape and would give it to the girl -- it was a girl -- to print. She had a habit of not stopping printing and she would print out the whole page even though we had messages that said END, FINAL END. So we changed that and wrote, THIS IS THE END. I SAID THIS IS THE END! GOD DAMNIT! THIS IS THE END! The girl called us over and said, "I'm afraid to go any further." [Laughter]

RM: So IBM was seeing what was going on.

OM: In the same old way -- yes.

RM: Were you saying to them, "Look, this is the situation, how about supplying this to us."

OM: Yes. We were doing that, but it didn't have any effect. They didn't really get that until we got

started on the 709, that's when we had the SHARE Operating System.

RM: After you got block time on the 704 in New York then, of course, the machines came out here. What were some of the developments that grew out of that machine?

OM: Okay, the development that grew out of that actually started before then. That was when Bob Patrick and George -- I forget his last name -- came out from General Motors Research. Apparently we had told him about our 701 operating and they came out and proposed that GM and North American engage in a joint venture to develop an operating system for the 704. This was a three phased operating system. You don't hear much about those anymore. It had an input editing phase, execution phase, and output editing phase. The input editing phase took the input, the cards, and converted them into binary. Similarly, instead of doing the conversion, the output of the program was written directly in binary and it was post proceeded by a conversion program. We had an off-line card reader, and we also had an off-line printer. That was the North American-General Motors operation system. It was distributed to a fair amount of

people. Convair used it. Several other people used it.

RM: It was a curious thing that General Motors would have come to you. What was their reason? Did they have a 704?

OM: They were going to get a 704.

RM: Did they have a 701?

OM: Yes. I believe they had the 701. I'm not sure. I can't remember for sure, but I think they had a 701. Bob Patrick and I had been slightly acquainted, passing hellos, when I was at INA and he was at China Lake. I don't remember exactly the events that led up to their proposal for a joint venture. I think we had talked about what we proposed to do for an operating system for the 704.

RM: Was this the first true operating system?

OM: I claim the 701 was the first true operating system.

RM: Yes. The North American one.

OM: Again, this is a tape system with off-line printing and card reading. This is also true of the SOS system, which is also a tape system off-line. IBM was never very strong on their drums.

RM: While you were doing this work were they, IBM, in turn doing anything in the operating system area?

OM: No. IBM did nothing with the operating system on the 704. The first thing that they did in the operating system was the SOS system.

RM: And that went on the 709. Were there any other important offshoots and developments from the 704?

OM: The 704. I'm trying to remember. The 704 had relocatable loaders. Another thing that SHARE rammed down IBM's throat was the Roy Nutt assembler, which he wrote at United Aircraft. IBM had written an assembler. I had always thought of United Aircraft as being a West Coast company. Because it was an aircraft company, and basically the computing that was done in southern California was aircraft, there was a very close relationship between United Aircraft and the West Coast.

RM: But United is East.

OM: United is in Hartford. We told IBM that we wanted Roy's assembler instead of their standard 704 assembler and that we wanted them to distribute it, which they did. The way SHARE worked was that people would submit their programs, essentially they were unedited, and IBM undertook to distribute them. It got to be pretty big by the time they got through with the 7090 days.

RM: Did anybody act as a clearing house?

OM: IBM actually acted as the clearing house.

RM: So they checked out the programs as well as --

OM: No. IBM didn't check them out. The person submitting them was responsible for checking them out.

RM: Then if someone submitted a program which was not truly debugged, he was going to hear about it.

OM: Yes.

RM: Did this happen?

OM: Most of the programs submitted were pretty good caliber.

RM: So the Nutt assembler was distributed. Were there any other 704 adventures?

OM: Well, PACT-1, of course. I'm trying to think. SHARE, which is a national organization, was largely dominated by the West Coast group. General Electric was involved. Both General Electric and Westinghouse were strong in SHARE and the 704.

RM: Oh, were they? Just using SHARE programs or did they contribute as well?

OM: They contributed as well. Don Shell of General Electric in particular.

RM: In what way did Don Shell and G.E. contribute?

OM: He wrote programs. Actually General Electric contributed to PACT-1A.

RM: Did they have a 701?

OM: 704. PACT-1A -- 704. There must have been some other things that happened in the 704 days. We weren't too far into the 704 before we started getting ready for the 709, it seems. Things came fast and furious. [Laughter]

RM: Can we get some dates here? For instance, when did G.M. and North American do their work on their joint venture?

OM: We had that from the very beginning with the 704 which must have been the spring of 1955. I was also involved with the 705 after the 704.

RM: Oh, were you. Tell me about that. The 705 hasn't really occurred too much in the West Coast stuff because it was the business data processing machine. What were you doing?

OM: I was in charge of both scientific and business computations. That must have been about 1956 or so.

RM: Okay. So you had to write completely different programs for the 705?

OM: Yes.

RM: Now was anything done similar to PACT and SHARE and these kind of cooperative ventures on the business side of it?

OM: Yes. Jack Strong started GUIDE. GUIDE was a comparable organization for 702s and 705s.

RM: I haven't got into GUIDE and that is probably a mistake on my behalf. Can you bring me up on that one?

OM: Well, we got the 705. SHARE had been so successful that Jack got the push to get a GUIDE organization. We were never as active in GUIDE as we were in SHARE.

RM: Were the same people involved?

OM: No. It was a completely different set of people.

RM: Can you give me some names and companies?

OM: Oh, that's harder, because I wasn't very close to GUIDE. I don't think I ever went to a GUIDE meeting. Very few West Coast people, although there are some who I just don't recall, were involved in the 705s.

RM: Then the reason Jack Strong was asked to do it was because he had had the experience with SHARE.

OM: Yes.

RM: Not because he was into the 705.

OM: Yes. Well, he was concerned with the 705 also.

RM: Were you writing 705 programs?

OM: Yes. I wrote 705 programs.

RM: What were some of the differences in program writing and problem solving that you had to consider from the 704 to the 705?

OM: Lots of them. The 704 was a word machine. The 705 was a character machine. There is really not so much difference in the programming as programming.

RM: But you had different problems to cater to.

OM: Yes. These were all business applications where the emphasis was even more on data manipulation.

RM: So the tapes, the peripheral stuff, was very important.

OM: Yes, yes. Basically we used the IBM programs like the assembler. We didn't really have a loader.

RM: Were there any 705 programs that you recall which carried some impetus that went through GUIDE or maybe through yourself?

OM: No. I can't think of any on the 705 that had influence outside of North American. We did write an asynchronous 1/o system for the 705, but that didn't have any influence outside of North American.

RM: In the case of the 705 business data processing, were the companies not as coordinated?

OM: That's right. There was very little communication.

RM: Was this maybe because their needs were different or were they competitive? Have you any feel for why this?

OM: I think they were used to being competitive. I don't think they were really competitive. They were

spread all over the country, but I think it was kind of a habitual way of approaching things.

RM: The West Coast computing environment was different because there was such a large collection of people in such a basically small geographic area.

OM: Yes, and they were all somewhat supported by the government. The result was they had a largely altruistic attitude toward computing; there was more concern with the profession than for their company. That seems to be true with programmers in general to a large extent, but I think it was particularly true with the people here in California. You heard very little about proprietary restrictions.

RM: It's fascinating because certainly in some cases they were competitive. In some instances the government had given two companies the same contract and said go at it.

OM: Oh, yes. But, I think there was a little bit of patriotism involved, back before it got out of style.

RM: So, in your view, the West Coast wasn't very influential in data processing developments.

OM: No. I don't think so. IBM took much more of the initiative in that area. I had the feeling they thought there was more of a market there.

RM: They might have been right.

OM: There wasn't much more of a market for the 705s than there was for the 704 and the 7090s. They kind of missed the market on them, underestimated the market.

RM: Underestimating markets has been a chronic case with the computer industry. What happened after the 705 work?

OM: That brings us up to the SHARE Operating System.

RM: Who were the people and what was this development about?

OM: Okay. The SHARE Operating System was designed by a group who met at Rand Corporation. Let's see if I can remember the members of our group. Don Shell was one of the group, in fact he chaired it. I was in the group, Irwin Greenwald, Chuck Baker was in it. There were two girls from IBM. I'm trying to remember their names. One of them was Elaine Boehme. They both came out of engineering programming. The other one was, I can't remember her name. Elaine wrote a big part of the assembler and the other girl wrote another big part of the assembler. Tom Steele was on the committee, Charlie Swift -- he is here now -- was at Convair at the time and was on the committee. That accounts for most of them. We

designed and wrote the specifications for the SHARE Operating System and told IBM to do it. One of our mistakes. IBM was late. It was buggy for quite a long time. It was one of the first classic under-estimations of programming systems.

RM: The time it would take to debug?

OM: Yes. How long it would take to program and debug.

RM: Did you, in this sense, tell IBM what you wanted, including asking for certain equipment or peripherals?

OM: No. It was an operating system only. We provided the design, they were supposed to code it and check it out.

RM: Was the first time that a computer manufacturer had put together an operating system.

OM: It was about the last time we dictated a design of a system, as far as I know. [Laughter]

RM: Was it a successful system?

OM: Yes. I think it was a successful system. It was a multiphase system, although basically it was used as a three phased system; input editing, execution and output editing.

RM: Was it distributed through SHARE?

OM: It was distributed through SHARE and used widely. It wasn't used as widely as it might have been because the FORTRAN compiler wouldn't run in that

system. They had a new FORTRAN for the 709, and the FORTRAN people were very smart. They took one look at what was going on in the other part of the programming and decided they were never going to be ready on time. So they have their own supporting routines. They never operated with the SHARE Operating System. There were two systems really. FORTRAN really wasn't a system, it operated much in the same way as the 704, but they found that the other people weren't going to get there, so it ended up with two sets of programs coming out of IBM.

RM: The customer had a choice.

OM: Yes. Most people ran with both.

RM: What were the advantages and disadvantages in one or the other?

OM: Well, one advantage is the language. Of course, obviously, if you wanted to write FORTRAN, you had to run it with FORTRAN. FORTRAN was generally easier to program.

RM: For instance, the company who had to train a whole slew of programmers might have opted for FORTRAN.

OM: That's right. At North American we programmed half and half, or maybe two thirds in FORTRAN.

RM: But the operating system was there for use.

OM: Machine language use.

RM: Now the 709, was that scientific?

OM: Scientific basically, it's a forerunner of the 7090. We used it for both scientific computation and for business computation.

RM: It was flexible enough by this point.

OM: Yes. We wrote our own operating system for the business applications based on the SHARE Operating System.

RM: The 709 brings us up to what date?

OM: The 709 was followed by the 7090, which really brings us up to when I left North American in 1961.

RM: The 7090 was compatible to the 709?

OM: Yes.

RM: Why did they quickly go from one to the other?

OM: The 709 was in the same architecture as the 704, the same components, it was a tube machine. The 7090, of course, was transistorized.

RM: So the 709 was the end of that era.

OM: That's right.

RM: Can you think of any other 709 or general developments that we haven't talked about that maybe we should cover?

OM: I'm trying to think.

END OF SIDE II

TAPE II, SIDE I

RM: I thought we could talk briefly, or not so briefly as the case may be, about the beginnings of Computer Sciences, how it got set up, the people involved in its charter.

OM: When SHARE was first formed, Fletcher Jones was working for Frank Wagner. He became the first SHARE secretary and he did a really outstanding job. After that time Fletcher went to Columbus, where he managed the Columbus Division for North American. In about 1959 Fletcher proposed forming the Computer Sciences Corporation. The second man involved was Roy Nutt from United Aircraft who wrote the SHARE assembler. Roy worked with IBM on FORTRAN. He was employed by United Aircraft at the time, working in IBM on the FORTRAN compiler. I knew Roy then. In fact I would see him when I went to New York to do checkouts. I would see him also at SHARE meetings. The third member of the original group was Bob Patrick. He was the guy I mentioned that came out of China Lake and General Motors Research.

They sold a contract to Honeywell to do the FACT compiler. That must have been a great job of salesmanship because at the time they sold the compiler to Honeywell, I'm not even sure they were incorporated.

That is an interesting compiler by the way, because in my opinion, it was a better language than COBAL. That was actually the forerunner, the results, of another language that Grace Hopper had done before that.

They started hiring people. They hired two from North American. Bob Paul who used to be at the North American RocketDyne Division. He is still with Computer Sciences. They hired Dale Hanks who used to work in the Engineering Department. He didn't work for Frank Wagner, he was in the Structures Department. They hired Charlie Swift who used to work for Convair, San Diego. These were the key people on the FACT project.

RM: What did FACT stand for?

OM: Darned if I know.

RM: You said that in your opinion FACT was probably a better language than COBAL. Can you tell my why you thought it was a better language?

OM: More flexible. More powerful. It was a little more symbolic. It was the result of an awful lot of compromises. I'm not a COBAL fan.

RM: What exactly did happen with FACT?

OM: I don't really know. It was implemented and used on the Honeywell computers.

RM: FACT was designed for the Honeywell machines?

OM: Yes, but the language was a machine independent language. There was one other competitor and that was the language that IBM had developed. I can't really remember the name of that language. Dick Talmadge was one of the California people who was very strong in the development of the IBM language. In fact, he even sponsored the group that implemented it. Dick Talmadge came out of Lockheed, California also. He was involved with the very early SHARE meetings.

RM: So out of FACT came the beginning of this corporation, Computer Sciences Corporation.

OM: That's right. The next thing they got was a contract to do a FORTRAN compiler for Livermore. They got Bill Gatt who used to be down at U.C. Los Alamos, and they also got Dave Ferguson who worked both on the FACT compiler and the FORTRAN compiler. Dave Ferguson was at UCLA before he came to Computer Sciences. He wrote the SAP assembler for the 709, which was a rewrite essentially of the assembly language that Roy Nutt wrote for the 704. Dave Ferguson since left Computer Sciences and formed his own organization. I can't remember the name of it.

It has since been bought by somebody else.

RM: Something that seems to have been occurring is that SHARE brought people together, but then by the late 1950s and early 1960s, people are doing their own thing again. For instance, the FACT project. Was that done through the SHARE organization?

OM: No.

RM: Could one say that by this time SHARE seems to have lost its usefulness?

OM: I couldn't answer that fairly because since I came to Computer Sciences, I haven't been active. So I don't know. Maybe I am representative. I --

RM: I was thinking about the period 1959 through 1961.

OM: I don't think that was true at that period so much. Still, I don't know. It was maybe because I was still at North American in that period.

RM: This corporation, for instance, got started and obviously has proliferated. The early work which could have perhaps been done through the SHARE organization, wasn't. I was just wondering if there are any conclusions to draw.

OM: Yes, there are some conclusions we can draw. There was money to be made in doing software for the manufacturers, and also for doing programming for other

people. Up to this point that wasn't done very much. Computer Science and Computer Usage were about the first two in software.

RM: In fact, that is probably a good comment about one of the reasons why software is so many years behind hardware. At first it wasn't considered even to be a necessity, then secondly it wasn't a business, and it was only when it became a business that things really got started.

OM: That is possibly true. It is a lot harder to understand than hardware. People still have trouble controlling and understanding software.

RM: Do you have any philosophies about why this generation gap?

OM: Well, you have got to have the machinery first. Software is more intangible, it makes it much harder to control. It is much easier to say you can't afford that when you are talking about hardware, but it's more difficult to say that when writing a few lines down on a piece of paper. We are not the only field where that happens. If writing laws was a commercial endeavor we might become more aware of how expensive it is to write laws. For thousands of years we have been writing laws, and haven't gotten much better at it.

RM: That is a nice analogy. Going back a decade or so, as a user and a programmer, were you trying to lay it on the manufacturer, which was mostly IBM, that they really should be building machines with the user in mind? Maybe even coming to the users and saying, "Look, what is it you need in a machine that we can give you?"

OM: By supporting programming, yes. One of the things we were preaching from the very beginning was that they should be doing this programming for us.

RM: Was it heard?

OM: Yes. I think so, because that is definitely the way that things are now. Not only IBM, but all of the manufacturers felt compelled to provide not only the hardware, but assemblers, loaders, FORTRAN compilers, COBOL compilers, PL-1 compilers, maybe sort routines, mathematical routine packages, input-output handling packages, and so forth. More and more that has gotten to be recognized as a very large investment in terms of producing a piece of hardware.

RM: Owen, thank you very much.

END OF INTERVIEW

Index - Owen Mock
12 April 1973