Computer Oral History Collection, 1969-1973, 1977

Interviewee: John Lowe
Interviewer: Robina Mapstone
Date: October 16, 1973
Repository: Archives Center, National Museum of American History

MAPSTONE:

Today is October the sixteenth, and this is Bobbi Mapstone, and I'm talking with John Lowe--L-o-w-e--and this is an interview for the Smithsonian Computer History Project. Okay, let's start off with how you got into the business. That's a good start.

LOWE:

I got into the business because I was a bookkeeper, and how I got to be a bookkeeper is not really part of the story, so we'll skip that. Am I coming through all right? Yes. But I actually worked for about two years as an old-fashioned, black armband and green eyeshade type of bookkeeper. Then in l94l, December of l94l, I was attracted by the enormous wages that were being paid at the aircraft, airplane factories. I applied for a job at Lockheed and was hired as a typist for 4l¢ an hour, I think it was.

MAPSTONE:

[Laugh].

LOWE:

And, after working for a few months as a typist, I got acquainted with some people who worked in tabulating, as it was then called. The word computing, as we now understand it, was hardly, indeed, had never been invented at that time. I was very much fascinated by the machines and the work I was doing; I enjoyed it; I learned quickly. The war, of course, had started by this time; things were very tight; manpower was short and the work load was heavy. Everybody was working full capacity and many, many hours overtime in these fascinating times. And that's probably one of the reasons why I learned as quickly as I did, as it was sort of a sink-or-swim proposition. At one time I was responsible for getting out the payroll for Lockheed. There were several occasions where we almost didn't make it on time, but somehow we managed to squeak through. And during this time--I would say in '42 or'43--I became friendly with Bill Bell, William D. Bell, who had even before then started doing engineering work on the tabulating machines. I started working with Bell in his engineering activities. As I remember it, the first job we set up was to reduce wind tunnel data. Later, we developed procedures for determining eigen values of complex matrices, which was done in connection with flutter

analysis in the engineering department. Bill also did a lot of work with a fellow named Crawford--I think his name was Ray Crawford, in the engineering department--on learning curve calculations. I can't remember exactly what this was about, but as I recall, it was a generation of a family of curves which were used in cost forecasting, the projected costs of future articles based on the cost of the first article. We also did some work on uniform or constant interval tables, and I think I showed you earlier a copy of a table of logarithms which Bill and I worked out in those days, based on the theory that a constant second difference would give a uniform error, uniform maximum error, for linear interpolations throughout the range of the table. For today's machines it's a pretty useless concept, but for that time, handling a card was so slow that it was important to keep down the number of entries in the tables to save a card in the deck. So we used pretty much the standard equipment in those days. I think I told you earlier the story about trying to get the minus signs put on the tabulator, which caused IBM to look at us as if we'd gone out of our gourds, but we finally managed to get it done. It wasn't easy. Also, at that time, and partially in conjunction, I believe, with Bill Bell, a fellow by the name of Tom Lothar, who was the IBM machine repairman, developed what was called split multiplication for the 601. Probably I'm one of the few remaining people who's ever heard of the name. But, without trying to be too detailed about it, it greatly increased the flexibility of the machine as compared to the unmodified machine.

MAPSTONE:

Did you do this using some kind of plug board for these methods?

LOWE:

Yes, which were the only ones extant at that time, the only way you could program a machine. I guess it was just about--not at that time, but some time later, it was developed, I believe, by a couple of Englishmen, whose [names] I can't remember now. Do you have a pause feature on the machine, where we can just stop the tape temporarily? [Recorder halted]. The rest of the equipment was unaltered, just the way IBM and God made it, with sorters, reproducers, recorders, tabulators. The tabulators were 405s. Lockheed, at that time, had one of the largest assemblages of tabulating equipment in the country. I don't remember the numbers anymore, but I know we had a very large room, like a barn or a warehouse, packed solid with equipment. So are there any details about this phase of things that come to your mind?

MAPSTONE:

One question is: Did at any time people consider using Rem-Rand machines?

LOWE:

Not at Lockheed, not to my knowledge. I never heard of it. This was considered, however, just skipping forward a few years, when I went to work at Douglas. At the time I went to work at Douglas, Douglas used Remington-Rand equipment exclusively for

tabulating. It was quite a political battle, as a matter of fact, to get the IBM equipment in the place. It was one of the battles fought by then-boss, Charlie Strange, of whom we had spoken before. As I remember, Charlie politicked for several years before they even let an IBM salesman in the door. It was considered a Burroughs thing. This persisted actually pretty much all the time I was at Douglas. We had very little if any communication with the people in the accounting part of the company. We used available equipment; which is too bad, really. It set Douglas back a long way, of course, compared to Lockheed, where really the computing was an offshoot of the accounting activities; I described earlier my own adventure. I believe--I know--that that attitude persisted long after I left Douglas in l96l. I understand that at quite a recent date they still had some Rem-Rand old equipment. The bitterness persisted.

MAPSTONE:

Do you happen to know what started the bitterness? Is there one specific fact that took place?

LOWE:

Well, I think it was simply empire-building and defensiveness. The people who had reliable equipment were having the whole show as far as mechanical, any record-keeping or computation, whatever there was of it. And they didn't want to have any interlopers coming in that might present competition for them. They were inordinately primitive, too. Having had considerable experience at Lockheed in seeing how payrolls should by run, for example, I found at Douglas their whole system to be just very, very bad--very expensive, very inefficient, very slow, with lots of errors. But I guess that's all beside the point. This came about, as we talked about. The kind of equipment at Lockheed, as far as I'm concerned was never considered. Actually, it was completely unsuited to computational work simply because the machines were too inflexible. They did not have a counterpart of the plug board or patch board. All their setup, they had a device somewhat similar, or analogous, I should say--in other words, ...a purely a mechanical device where connections between the whole sensing mechanism and the counters or other mechanisms inside the machines. And to alter all this, the structure, was a factory job. So that is the reason why Remington-Rand--probably the reason why Remington-Rand was so far behind IBM in the computational field.

MAPSTONE:

I didn't realize that. Somehow I just presumed that Rem-Rand also had gone the way of the plug board, but that's not true. Did they get into the plug board later, or did they go straight from punch cards to computers?

LOWE:

If there was ever any plug board equipment, I don't know about it. I think they went right to computers. And of course they bought their way into that by buying Eckert and

Mauchly and a couple of very small computing outfits. Last I knew, they still were way, way behind, and probably it all goes back to that one simple little fact in the beginning--that their machines simply weren't suited to computational work.

MAPSTONE:

So really the plug board was an incredibly significant development.

LOWE:

It was the development. There's no doubt at all about that. As late as l955 or '56 there were still proponents of the plug board as opposed to the internally stored programming machines. This was a view espoused strongly by some people at Northrop in particular. [Chuckle].

MAPSTONE:

The wooden wheel people?

LOWE:

Mhm, right. Greg Toben and Bill Woodbury and a fellow who—

MAPSTONE:

Rex Rice.

LOWE:

Rex was the one I remember having lunch with. He had gone to work for IBM early on, and he built, at his instigation was built, a plug board programmed, if you like, machine computer. And Rex came out particularly to interest me in this machine. And I just said, "No, Rex. ... I know all about plug boards. I've got calluses on my fingers from wiring them, I still have calluses in the palm of my hands, that have never completely vanished, from pushing wires into boards." [laugh]. But I said, "Time is passing by." And he didn't like this. I remember he took some umbrage at this benighted view of mine. [laugh].

MAPSTONE:

He was a strong advocate of the plug board concept. The wind tunnel data that you were working on, was this part of this West Coast cooperative wind tunnel venture?

LOWE:

Yes. And also Lockheed had a small wind tunnel of its own on the premises. But we got data from both of the two. Well, actually, no, that's not so, because the co-op wind tunnel

came later. No, at the time I'm talking about, this was strictly for the tunnel that Lockheed had, a small, unpretentious tunnel. The co-op tunnel didn't get into operation until after World War II, as it were. So I was wrong about that. I guess where I was mixing myself up--I did reduce data for the co-op tunnel for Douglas which was also one of the co-opters of the tunnel project.

MAPSTONE:

So were there any other activities that we--we talked about the wind tunnel and I'm trying to think of the other thing--anyway, some of the other activities going on at Lockheed that would have used the punch card machines?

LOWE:

Well, probably the chief one in terms of volume of computation was when I mentioned computing eigen values of flutter matrices of complex elements. Even with the tabulating machines it was a lot of work. I have wondered how they did it. Of course, I know the answer to that. [Chuckle]. When they did it by hand with the Friden calculators, they probably did a hundredth as much work as they asked us to do, which is recognized, of course, which is a familiar pattern that seems to persist unto the present time.

MAPSTONE:

Right. Quantum leaps.

LOWE:

And a little afterwards, you might say that people's appetite for computing is like a gas. It always expands to fill the available volume.

MAPSTONE & LOWE:

[Laughter].

MAPSTONE:

I like it! Were there any other people that you worked with in that early blossoming period whose names we should perhaps discuss?

LOWE:

Well, one of them was a man you mentioned the other day--Bill--Lynn Couret, for just a little bit, although as I recall it, not too much. I think in particular he was working, as I remember, pretty much strictly as a machine operator. I don't think he ever applied himself to learn some mathematics and the theory of what we were trying to do.

Essentially, I think it was Bill and myself with just some, shall we say, clerical level help in terms of--that's not a good word, either, meaning routine, people that do the routine work of feeding the cards through the machine et cetera.

MAPSTONE:

Yes, what today would be operators.

LOWE:

Machine operators, yes.

MAPSTONE:

Where did Ward Beeman fit into the Lockheed picture?

LOWE:

Ward was an executive at Lockheed during this time. I don't remember what specific job he held, although it was in the engineering department, I remember that much. Well, you've talked to Ward probably, haven't you?

MAPSTONE:

No, Ward's in Australia.

LOWE:

Oh. Well, I kind of think he was in charge of the strength analysis section, or stress analysis section. That's a half guess. And I didn't know Ward particularly well. He and Bill became quite friendly. I never did get to know Ward very well. I don't mean to imply anything negative, that is, it's just I wasn't exposed to him very much, Bill being pretty much the front man for the two of us. He was the one who mostly dealt with the engineers, the customers, and the like. I was more or less the inside man with the skunk work.

MAPSTONE:

Bill made out okay.

LOWE:

[Chuckle]. Bill was, and always was, an expert salesman. He was a very good one. In those days of course, that was extremely important. Just to get people to want to try, to dig a little money in to try to see if you could get somebody to work on a machine--it was a very revolutionary idea!

MAPSTONE:

That's right. What about, just to get some relationship to what was going on at Lockheed, what were some of the planes and contracts that were happening during this period, 1941 to what was it, about '45?

LOWE:

Yes. The Constellation, of course, was being built. It was a military transport. The Lightning, the twin-boom P-38. The Navy had a submarine patrol plane, the P2V it was. I don't know if l can remember any others.

MAPSTONE:

All right. How about moving on now to your short, happy stay with your little service bureau? That was the next jump..

LOWE:

Right. We have the name of that service bureau on a sheet. I'll go get it. Don't worry, it's unimportant. It was essentially a man named Charles W. B. McCormick who had rented two pieces of equipment, a 601 multiplier and a keypunch and had gotten a contract to do some orbital calculations. When I went in with him, he had been fiddling around for some time and productivity had been zero. So he sort of said, "Here, John, you do it." And wired up--I would guess, I don't exactly remember--probably about fifteen boards for this 601 multiplier. And we managed to crank out orbit calculations. We had a lot of extemporization and some pretty crude procedures, but we did it. We also did a few other things which I can't remember in any detail. I was only with Charlie a matter of months really, and I was offered a job with Douglas, and went to go with Douglas in, I believe it was April of l947. Vaguely, it seems to me I was with Charlie from like October of '46 to April of '47.

MAPSTONE:

And McCormick didn't stay in business?

LOWE:

No. He folded up, oh I would say right within a year after I left him.

MAPSTONE:

Well, when he folded, did you get the feeling that an engineering service bureau really was going to make it or that there was a need for it at this point?

LOWE:

Well, I guess I'd have to say no, because if I'd had any real confidence that it would go, I probably would have opened one for myself. Maybe Christman Computer Sciences Corporation is a few years ahead of its time. [laughter] I think it, probably, would have gone. Charlie had a really good idea. His operation, I think, fell apart because Charlie was simply, while a very delightful guy, probably one of the world's poorest businessmen. He, among other things, was chronically late. He would make an appointment to be at a customer's office and be two hours late for the appointment. [laughter]

MAPSTONE:

And no matter how good your product—

LOWE:

But I think it could have gone. Oh, of course, again just about this time Telecomputing got started on a similar basis, although they had enormous advantages of a captive customer, namely Lockheed. But they certainly prospered some long time, at least.

MAPSTONE:

Yeah.

LOWE:

If, I think, they had stuck to their original principles and premises and so forth, charging off into, [not] so many blind alleys, I think they could have prospered very, very nicely.

MAPSTONE:

Do you think by this point in l947, that your customers had caught on to the fact that these machines really could take care of their engineering calculations?

LOWE:

It was beginning to be known, although it was still a very novel idea. Among other things, it was very hard to present this idea to somebody who would say, "Well, tell me how your machines can do my job." It's sort of like somebody trying to explain a beautiful sunset to a person who is blind. You have to first try to tell him about how your machines work. This is just not practical in a short period of time. So it was a very hard project to sell to the average person. Now, of course, a lot of people weren't average. They caught on right away and that's what made the thing grow. Oh, Charlie McCormick, to begin with--despite his shortcomings--he saw many possibilities in it, to the point that he put his money on the line to try to make it pay off. Many people saw it.

But by and large it was quite difficult to get the point across.

MAPSTONE:

Yes.

LOWE:

It was difficult even within a company, again Lockheed or Douglas. And if you did it cold turkey--that is, when you approach a man who doesn't work with you or for you--why, it is much, much more difficult.

MAPSTONE:

I suspect that one of his problems was the fact that management was not engineering-oriented, dealing with people who perhaps didn't quite understand the basic concept there.

LOWE:

I think that's probably not true of the aircraft companies. Their managements were typically engineering-oriented. For example, Donald Douglas was initially an engineer, and trained his men there to work this way in the early days of the company. Jack Northrop was an engineer. The head man at North American, whose name escapes me at the moment, was trained as an engineer and worked his way. So managements, I mean, after all, the running and the producing of aircraft is probably more of an engineering job than it is anything else, except money. [Laugh]. So I don't think that was part of the problem. And then you ask why, really, the big aircraft companies were the, shall I say, the birthplaces of computing in this country, largely--I mean, in the realistic scale at least.

MAPSTONE:

Certainly in computing, yes. Very much.

LOWE:

Do you see any more of the names, as I do right now, of the people who got it started in the several companies around Southern California? Though it was probably not they, it was probably a fair evaluation. But even then it was still difficult. That's probably why Bill was as much of a success as he was. More than his technical skills, which were not inconsiderable--I don't mean to imply that--except that he was a darned good salesman.

MAPSTONE:

Which was really, really needed in those early days.

LOWE:

Exactly so. It's where Bill got ahead of a lot of people in this whole business. Let's face it, the typical technical man is not a good salesman. To find a combination of the two was probably again what we, with Bill, picked right up.

MAPSTONE:

Yea, the foresight, because so often the people who come up with the idea and the venture, so to speak, tend to become carried away with [their] own idea. He can't even conceive that the rest of the world can't possibly understand it and not jump on the wagon right away.

LOWE:

I've been guilty of that more times than I remember! [Laughter]. Exactly so.

MAPSTONE:

All right. So, in April l947, you left McCormick and joined Douglas. Let's sort of just take a look at Douglas, what was happening at Douglas at this time, the kind of computational work that was going on in this.

LOWE:

Well, we earlier touched on the difficulty that Charlie Strange, my boss, had in getting equipment into Douglas. So we needn't go into that. Another somewhat interesting thing is that historically the month I was hired, April '47, was the nadir, the low point, of Douglas' fortunes, the [? score] starting prior to World War II. After World War II, the aircraft business fell on perilous times. They had greatly scrimped with the whole operation. There was no market, or very little market, for commercial planes because the government had dumped all its surplus military planes on the commercial market. So times were tough at Douglas, and I remember very well Charlie telling me that this venture might not turn out to be one that the company could afford and I should be careful before accepting this job because he couldn't really guarantee I was going to have it too long. After all, we're spending something like a thousand dollars a month for equipment and my salary, which was something like four hundred dollars and this was a lot of bucks for Douglas at that particular time. [laugh]. He was very serious about it. And for quite some time I was the person who was in these machines. I worked a lot of hours on it. I remember they scrounged around to find a place for it for reasons which escape me now, and which aren't important, we had a little corner, a little room opening off a corner of the blueprint vault. I guess--I know now, because the blueprint vault was air-conditioned, and we felt that it was important that the machines be in an air-conditioned atmosphere. Well, it was a room, gee, I would guess like fifteen by twenty feet. [laughter] We had a sorter, a collator, a tabulator, a multiplier--correction, we had two multipliers. We started right off with two of them, because somebody had

convinced them that it was most important to have two, one to check the other, which was indeed true. All of this, of course, was ordered before I appeared. And an alphabetic printing keypunch, and I believe that was the list. I know, as soon as I went I started agitating to "let's rent a verifying machine." It took me months and months and months to convince the management that this money, which was like, oh, twenty or thirty dollars a month, was an appropriate expenditure. [laugh]. And I got awfully tired of making lists and checking them back against those tabulators! [laughter]. Charlie's secretary did most of the keypunching. I did some. Anyway, we went along and prospered awhile. I guess something like a year later, we hired a girl to do the keypunching and help them run the machines, the sorter and things like that. Her name was Nadine Bearley, and she retired from Douglas a few years ago. She'd been with them, I believe, for over twenty years. Well, anyway, that's sort of the atmosphere that existed at Douglas around then. [laughter].

MAPSTONE:

Now this was before the RAND split off.

LOWE:

Yes.

MAPSTONE:

So you were doing Project RAND work as well as sort of mainstream Douglas projects?

LOWE:

Yea. Notably, as far as I can remember, in helping on the preparation of the rather well-known RAND file of random numbers. As I remember it, we had twenty thousand cards, each with fifty random numbers in it, which would be, what, a million numbers? Yes. A million digits.

MAPSTONE:

Yes, that's right.

LOWE:

The actual--the pile had been prepared by a special device before I went to the company. As I recall it, this device essentially integrated the number of particles entering a radioactive source in a given period of time and translated it into a digit, which was chalked out on a keypunch. One of the things that I did was really extensive tests for randomness on this file. That was purely acting as a slave unit; and we didn't know what this was all about. And from the results of these tests we decided that there were some biases in the file, and we did some things to further randomize. I remember one in

particular was adding these cards in pairs, two successive cards added without carry, which was a little challenge in those days. I cursed that thing. It would have been so easy to do them in forty-column punch cards but some dummy had put fifty in. That made the challenge.

MAPSTONE:

Yeah, I bet.

LOWE:

So I also did some work at that time with Cecil Hastings in his proof reading projects. I don't now remember any other work I did for them, although there was other work. Anything else? No.

MAPSTONE:

Okay. So shortly around this period you must have started to get to the CPC. And I was wondering about what significance the CPC made.

[Recorder off]

Okay. As I was saying before, this is the time when the CPC came in, and I just thought it might, I'm just wondering about the kinds of differences the CPC made to what you were doing, the impact it had.

LOWE:

Well, actually you skipped sort of a generation, if you like, here. And that was, first of all, there were two successors to the 601 that came. First there was the 602, which was a really terrible machine, very unreliable, and it was shortly succeeded by the 602A, which was a very good machine. We used those extensively. For this type of operation that was going on then, there were about five or six of us. And then came the 604, which is the first, shall I say, as far as our work, at least, commercially available electronic calculator. It had vacuum tubes, it'd deal with multiplication with electrons.

MAPSTONE:

Was that not the 603, the first one, I believe? Looked like a big suitcase?

LOWE:

There was a 603, but as I remember it was only experimental. I don't think that there were ever--I think there were few, if any, of them in use. There was later, considerably later, a 605. It was around, actually, until quite recent times for certain applications. But the 603, if I'm not mistaken, was not commercially available. I think it was sort of a

prototype for the 604.

MAPSTONE:

I had a feeling that the 603 was. You know what I'd like to do? [break in interview] Well, anyway, we were talking about the first electronic calculators which were kind of the predecessors to the CPC.

LOWE:

Well, the commercially--at least in the CPC, I'm quite sure that the only machine used was the 604.

MAPSTONE:

That was the official CPC, yeah.

LOWE:

Right. It was the one they sold. That was the ... machine.

MAPSTONE:

Right, right.

LOWE:

That came in, about '50 or '5l?

MAPSTONE:

Probably. It was announced in late '49 or '50, and probably didn't make the market until after that.

LOWE:

I remember in one of the early CPCs the storage unit consisted of a mechanical device, which was--actually the registers were abstracted from the 602A, and they were completely identical. We had--was it sixty or was it thirty-two storage units in each box, which was called an icebox because that was exactly what it looked like. And then in later developments of the machine you got two of these iceboxes. Heavens, whoever heard of such storage? I remember they had one at the old Institute for Numerical Analysis at UCLA, and some wag had put a sign on one of the boxes which said, "Space." And then on the other box it said, "Hilbert Space." Crazy. Of course, the only problem we had with the CPC was learning how to use it, the idea that it was--and it was a rather hybrid machine. I think even today people would find it difficult to program for.

It had the rudiments of a selectable program. You could select one of two, or even one of three or one of four alternate programs that follow in a particular juncture. I remember well the arbitrary decisions I made when I set up our system for it. I limited it to two because early on I was sure that people were going to get so fouled up that they would waste more time than they could ever conceivably save. I think it was a good decision. Anyway, at RAND we started with, we had some of the early CPCs. I think we had three or four of them actually; at this time the operation started to get larger. One perhaps interesting sidelight: as long as I was at Douglas, I retained a couple of tabulating machines, ordinary punch card machines, including the 604. Even up to that time, which was '6l that I left, it was my firm conviction that there were a lot of jobs which could be done easier and less expensively on these standard tabulating punch card machines than you could do even on a 701. So that was one decision made back in the beginning. Even though we got the CPCs, we didn't completely turn loose from our old ways. Some jobs, sequential jobs, of course, were the great boon of the CPC, jobs with the integration of differential equations and how it verifies it, one technique which, of course, has many, many ramifications. But where the result of one calculation is used in the following calculation, there the CPC is the ideal machine, despite its limitations. But it was a strange thing [doing] a program for it. It was hard! Just plain looking back now, if I had to do it over again, I'd still think it was hard.

MAPSTONE:

Would you? [laughter] And there wasn't very much guidance.

LOWE:

No, not really. Not much of any in those days. That was before IBM had assumed the Great White Father attitude to at least the full extent. They were providing some semblance of guidance. Of course, again, they were largely being guided by myself and others like me in this area. We were, I think, without any question much further ahead in how to use these machines for computational purposes than anybody at IBM, or probably than anybody elsewhere in the world. I think we had an interesting little situation here in Southern California for about ten years. So, oh, for example, there was--the CPC, as of course you know, the individual units of it were plug board-controlled. There was a lot of schools of thought about the optimum way to divide these plug boards. I mean, look at these. You could wire a plug board for every job if you wanted to, which, of course, is ridiculous. But there were very many gradations of variability. You could have just one plug board which would do everything, or you could have half a dozen different kinds of jobs. So it was a considerable subject of discussion. Well, anyway, the CPC didn't last too long. I guess its total lifespan was two or three years, and I think its demise was for the greater good of the industry. But it was an interesting device and a lot of good work was done on it, but it simply was one that I think we all are glad is dead and buried. So then came out the 701, which I think we got in l953. IBM originally made, as I recall, a batch of twenty. I'm sure I'll be able to correct things, that you probably have heard many times over. And we got one of that first batch of twenty machines.

MAPSTONE:

They only made, I believe, a sum total of nineteen.

LOWE:

Maybe that was it.

MAPSTONE:

There might be one more.

LOWE:

Well, twenty sticks in my mind, but I surely couldn't prove it. And now, of course, we soon recognized the enormity of the task in front of us. How were we going to deal with this thing? It's one of those things you never really understand until you get in and work with it and you're faced with day to day problems. Here's this darned machine and here's a problem and how do I get the two of them together?

MAPSTONE:

$64,000 question.

LOWE:

We also, I guess fairly early on, realized that it was stupid to continue to reinvent the wheel by ourselves. Pretty early in the life of the 701, I guess we got one of the first meetings of this kind again at Douglas. I invited the proper people from the several installations where this first nineteen or twenty--I don't remember when it was. But we had, I think, a three or four-day seminar, which Douglas hosted--I don't see any papers about it here--where we discussed ideas and problems, and proposed actually some joint efforts in the programming field; at least one thing that did come out of it was we exchanged information about what programs we had. If anybody, in theory, if anybody were particularly interested in this particular subject and could ask in more detail, we decided right off the bat it was, well what SHARE tried to do later, which actually was to publish the programs and distribute among everybody. It was impractical. And I guess SHARE decided it was indeed impractical. And that kept up for a fair while. We had other meetings. One in particular I remember at, I think it was, at Northrop. We also had in those early days of the 701, we, in conjunction with RAND and also with the people from China Lake, the Naval Ordnance Test Station at China Lake--Harvey Tillet was the boss man, whom I think I mentioned earlier today--the three of us went together and developed a programming system for the 701 which we called PACT--P-A-C-T--which is an acronym for something I don't remember anymore. You've heard of it before, I take it?

MAPSTONE:

Yes.

LOWE:

A fellow named Charlie Baker was sort of the master-mind. He worked for me. That was successful, and it was actually used quite a bit, I believe. As a matter of fact, in l963 I was doing some consulting work for RCA, for Cherry Hill, and a fellow there told me--I think it was Grace--that PACT was then still being used on a Naval installation. They still had a 701 somewhere on the East Coast, I forget which one. A good ten years after it was developed.

MAPSTONE:

Amazing. Talking about the programming of the 701, when IBM first came out with the machine, I believe they did bring out their own programming. Ahm. I can't remember the name. Speed Code is the name that comes to mind. And I'm wondering why the different companies, specifically the West Coast companies, decided to develop their own programs. And I know you, for one, did. I was just wondering if you recall what your reasoning was.

LOWE:

Well, actually, the Speed Code, I think, came quite a bit later. I think when the machine first came out, IBM provided, it seems like input/output programs and that was about it. I don't think they provided what you would call a programming system. If so, it was so insignificant that I've just completely forgotten it. I don't remember it. But early on, long before we got the machine, we developed a programming system, completely apart from this PACT thing, which was in the form of an interpretive program. We had a basic machine language, coding system, and a system of storage utilization. It was a whole package of coding sheets of storage allocations, input/output programs all put on tape, library subroutines on magnetic tape which could be called in. The program store ... , lots of program between the drum and the core, or initially between the drum and the CRT. And everybody, I believe did that. A couple of people used ours, notably China Lake, and I think at least a couple of others--ours, or something essentially the same as ours or based on ours. But I guess, in answer to your question, I think that the help provided by IBM at that time was negligible if not nonexistent.

MAPSTONE:

So you really had to develop your own program?

LOWE:

Yes, yes. Even as early as that there was talk about, "Well, gee, let's get together and . .

." I guess I espoused the idea. I said, "Let's go together on a co-op wind tunnel. Let's each put up some money and we'll hire somebody to build us a program, under, say, a corporation like the ones that are already in business of corporate word directors, et cetera, et cetera. And we'll just hire this work done. That way, we'll get out of the squabble of each other trying to, each trying to tell the other, how to do it." And it might still, even today, I think it wasn't a bad idea. I pushed that same idea when SHARE came along and started trying to do its cooperative thing. Yes, but of course, it's sort of like that letter. I think it never did work very well, as far as I know. Of course, then they passed on from there to the fact that IBM should provide the programming. Again, as far as I know, at least until quite recent times, that never worked, either. Or it never worked very well. There were some obvious flaws here in the logic, it seems to me, expecting either "Let's get together and see this," ...sort of, type of thing, or, on the other hand, looking to the Great White Father to fill your needs. Anyway, we developed our own and it was a good system, I think. I speak with somewhat paternal affection. [laugh]. We did a lot of work with it.

MAPSTONE:

I'm sort of interested in the transition from something like the CPC and suddenly you're presented with the 701. Where did you learn, how did you get the ideas and concepts? Was this something that you could read about in a little manual that was sent out, that these kinds of abilities were available on the machine? Did people go to other programming schools? Had you, for instance, learned about UNIVAC? It's curious, how suddenly, you know, you're presented with this monster. And people seemed to know what to do with it. And I'm not quite sure why.

LOWE:

Well, I think part of the key to it is that many people like I at the time were pretty fascinated with the thing. It was a hobby as well as a job. Speaking for myself, it was an all-consuming interest. I used to spend all my time, all my waking hours practically for months working on this project. And now I guess you'd better change the tape.

MAPSTONE:

And now I guess we'd better change the tape.

[End of Tape 1, Side 1]

MAPSTONE:

Yes, we were talking about, or I was asking you about just how did you come up with a way to use the 701.

LOWE:

Well, I guess another salient fact is, taking the history of Douglas, we started in '47; just about six years later we got the first 701. Within that time we had gone through the 601, the 602, the 602A, the 604 and the Card Program Calculator. That's five machines in six years. So while, of course, the 701 was another large step forward as compared to these other steps, the idea of getting acquainted with new machines was not exactly novel to us. We also were entirely accustomed to depending on our own resources. We hadn't gotten any help from anybody that is to say in terms of how to use the machines. On the other hand, as I said, I think it's fair to say that we and others like us in this area were the ones to who people came for the answers. We were acknowledged as the leaders in the field. And that probably didn't really change until some time after the 701 came out, I'd say '55, '57, when the whole computing thing really started to explode and it became thousands where it had been hundreds of people. Still, looking back, I can't remember being particularly overawed by it. I was impressed by the new capabilities it offered, but there never was much doubt in my mind on how to go about using it.

MAPSTONE:

One thought that just occurred to me while we were talking: probably by this time you, the users, had become prepared for the kinds of things you wanted in a machine, and the 701 actually gave you some of these things.

LOWE:

Yes.

MAPSTONE:

So it wasn't like you were being bombarded by these brand-new ideas that had come out. You knew that you wanted storage, you wanted tapes, you wanted drums and interconnections; and then it's just a matter of how to do it.

LOWE:

Although I've always felt that IBM deserved a lot of credit for building the 701. It cost them a lot of money, relatively speaking, and I was told once by no other authority than Tom Watson, Jr. that it was a decision that was taken after considerable soul-searching and with considerable trepidation, that they weren't at all sure they were going to sell those first nineteen or twenty, whatever it was, machines, or that if they could sell them they would stay sold, that there was really enough computing to be done and soak up all that capacity, or if people would indeed by able to cope with the intricacies of the machine to exploit particularly. So I've always felt that IBM deserved a pat on the back for that decision. It was really probably the important one as far as modern computing as we know it is with us. It's what you call a milestone or a benchmark or a turning-point.

MAPSTONE:

Yeah.

LOWE:

It would have been taken eventually by somebody, because the technology existed. But it might have been years before, I guess, IBM did it, if it hadn't been done at that particular time.

MAPSTONE:

Yes, because UNIVAC was out, but—

LOWE:

Well, it was out on the line in a very halting and crippled sort of way. I guess UNIVAC in that form was never really a successful machine. It had at that time in particular, a mercury delay line memory, which--I never worked with UNIVAC, but based on the people who told me, the people at the Census Bureau--it was a real sure...to keep functioning properly. And those magnetic tapes were very unsatisfactory on the delay line.

MAPSTONE:

Metal instead of—

LOWE:

Well, that didn't last too long. It finally went to plastic tapes. They realized that finally, I think within a year, that metal tapes simply were an obvious mistake. So anyway, you asked a question I guess I gave a long-winded answer to. I don't even know what the question was anymore.

MAPSTONE & LOWE:

[laughter]

MAPSTONE:

Well, really it was around how--which I think you've answered--about just how did you get to view the 701, what was the way that, you know, I'd asked, you went from the Dark Ages suddenly to this New Age. But I think you've answered it. You really didn't; it was progression.

LOWE:

Yes, the Dark Ages really weren't so dark. To repeat something I said earlier, in some ways the, I think CPC was more difficult to program, a more difficult machine to exploit than was the 701. It was a son of a gun!

MAPSTONE:

Yes, I gathered from what you said and from what other people have said that it took tremendous creativity and ingenuity to make the CPC work. And I wonder if some of the people who were in love with the CPC weren't in love with it because they had figured out the way to make it work, and there was a tremendous sense of satisfaction.

LOWE:

I think the difference between them and some of the others of us--they were hobbyists. In other words, they were totally mesmerized. They were espoused to this idea of the cord program or the plug board program or, try it a third time, the plug board-controlled computer.

MAPSTONE:

Yeah.

LOWE:

They, I think, made themselves a little foolish after awhile.

MAPSTONE:

Well, somebody made a comment which maybe fits in with what you think, and that is that most or some of these machines were fine for the geniuses of the world, but the world is not full of geniuses, it's full of rather fairly bright and intelligent people.

LOWE:

And I don't even think all of these geniuses want to make a career of programming computers. [laughter]. There's something, sort of a concept or precept, that I try to keep in mind all through my computing career, that it's all too easy to get fascinated with the tools of the business as again perhaps--not that I mean to derogate them--but again there are just the [examples] in this discussion of the Northrop boys and the plug board. They became enamored of this tool, and I've always felt that it is very, very important that all our trappings and all our equipment are just tools to get some answers. As long as they serve that purpose, fine. When they don't do that anymore, throw them out. [laugh].

MAPSTONE:

Mhm. But conversely, don't throw them out until they've grown out of their usefulness.

LOWE:

Exactly.

MAPSTONE:

Which kinds of brings us to the story we were talking about earlier, about bringing in the 704.

LOWE:

Yes, which was a step in the march of progress we skipped at Douglas. Of course, now, by this time, which was '55 or '56, I guess, Douglas was no longer one. Douglas had several computing installations. I had only one.

MAPSTONE:

Which one?

LOWE:

Santa Monica Engineering Department computing facility. I think that's accurately its designation. And some of the others did go 704. But they had perhaps a better reason. They were new, and they didn't copy the procedures for the programs, the techniques that I had set up. Well, they did to a degree, but they were new, they were independent, they wanted to show their mettle and they didn't just seem to follow old granddad. So they developed new things. Also, of course, at that time you couldn't get 701s anymore. There were no more made, and the only way you could get one was when somebody discontinued one. And even then IBM was loathe to re--to let them out again into the field; they wanted to scrap them rather than deal with obsolete machines. So some of the other Douglas facilities did indeed get 704s. I don't remember who or how many anymore, but mine didn't. And again, in the context of your comment, I think it was a good decision because here was a case where I felt the extant tools were doing a good job and the advantage of scrapping the machines and changing them for new ones by no means was worth the cost. It's expensive to change computers. I guess people are finding this out over and over these days. [Chuckle]. Of course, nowadays it isn't so bad because they have COBOL, and they have FORTRAN and most programming is done in such a language. And they are, to a degree at least, universal now. That's been something that people have been trying for over ten years to achieve, progress, I guess. Well, in the days I'm talking about, which were the 701 and the 704, there was nothing that even resembled a common language. It would have been starting all over again with new programs, training the people again. We'd have to do something similarly on the 701. There was a very large body or library of programs that were written in machine language for the 601, 701, excuse me. We did finally. When we finally went to the 709--or 7090--we even then wrote a simulator for the 709 so we could run those old 701

programs. And that worked all right, too. That was written for us, by the way, by Computer Sciences on a contract. It was kind of interesting to us that it was about a break even in time that these programs were run simulating on the 709 about the time they were run on a real machine and being used on the 701. [laugh].

MAPSTONE:

Yes, that is interesting.

LOWE:

Obviously, this is to say that the simulator was by its inherent nature very inefficient. And the last I heard, that was still in use at Douglas. Some of those old 701 programs were still hanging around. Old programs are like old soldiers, they never die.

MAPSTONE & LOWE:

[Laughter]

MAPSTONE:

They never even fade away.

LOWE:

No. They just sit there and stare at you. [laugh].

MAPSTONE:

Well, going back a little bit to sort of the 701 era, when IBM came out with it they also made it possible to go to school and things like that. Were you involved in those activities?

LOWE:

Well, the nearest thing to a school--we had a meeting, I remember, in Poughkeepsie shortly before they delivered the machine, we being its prospective customers. Before then, some of the engineers explained to us how the machine worked. To the best of my knowledge and memory, it was some long while before there was anything like a manual for the machine, that is, some long while after the machine was delivered. There was a guy named Joe Smith who came out and gave us a series of lectures about how it worked, so it turned out that I, because of that meeting, I knew more about the machine than Joe did. [laugh]. So he wasn't a heck of a lot of help, but he was a really nice guy, I still remember him. No, you really sort of learned by listening and asking questions, it wasn't a matter of being handed a manual and saying, "Here, read it."

MAPSTONE:

Because there was something that the IBM customer engineering department [?] the 781, which supposedly was the first course for the 701. And this is probably the thing that you went to at Poughkeepsie. And Charles Baker, he went with you?

LOWE:

Yes.

MAPSTONE:

Right.

LOWE:

And a fellow named Rusebach, also. This would have been in l953, early '53. Yes. I was elected president of that class one year.

MAPSTONE:

Were you?

LOWE:

Yes. How about that? [laugh].

MAPSTONE:

I'll be darned. Valedictorian as well?

LOWE:

Sort of. I got up on graduation night and gave a little speech. Maybe that was that picture I clipped from the old IBM newspaper. I bet you that was. Yes, it sort of comes to me. We were lumped in with some machine repairmen.

MAPSTONE:

I was just going to ask you about that, because I, somewhere came across a piece of paper and I thought maybe that's what it was.

LOWE:

Where did you get that scoop? Did you talk to Chuck Baker, for instance?

MAPSTONE:

I don't know where. I probably got that from, well, North American most probably. Here's the list of the people that went to the first one. It's hard to say. How about your involvement with things like SHARE and ETA, that kind of organization?

LOWE:

Well, SHARE, I went to one of the early meetings, but I was never a SHARE enthusiast or anything. You saw a little sidelight on that in the letter exchanged between Paul Armer and myself that I read to you earlier. I felt SHARE was barking up a wrong tree in trying to export, essentially, its programming from IBM and trying to do all this. I went to some of the meetings and frankly just got terribly bored, listening to the same old bickering back and forth and regurgitation of the same old arguments.

MAPSTONE:

Did you feel that the cooperative process just wouldn't work?

LOWE:

Yes, I felt that.

MAPSTONE:

Because of the human element and too many people in different things?

LOWE:

And unfortunately the guys who had the really good ideas were not necessarily the ones with the loudest voices. You've seen this in organizations before.

MAPSTONE:

Right.

LOWE:

In fact, I'm sure a lot before then.

MAPSTONE:

Yes.

LOWE:

It was a democracy with no real leadership. You know, it was always using the eye-to-eye and hold-hands approach. I suppose SHARE's come a long way. Now I went with SHARE I suppose even as early as l959 or '60. I left Douglas in '6l, so it was prior to that. Again, I felt, as I think I mentioned once before, that the proper approach to some of these cooperative ventures was to form a corporation, have each member put up some money, and have the corporation hire people to do the job or contract to do the job or somehow get it done. But this somehow or another was never even seriously considered by SHARE. I don't think I ever won a single adherent to this concept, although I still think it's a pretty good one.

MAPSTONE:

You still feel that people should take care of their needs individually?

LOWE:

Oh, no. That's completely impractical nowadays. I mean, the cost of some of these operating systems is more than General Motors could afford. There has to be some kind of cooperative approach. And the way it's working now, of course, IBM and its competitors are supplying the programs, but for a charge. I don't know how good the programs are. My last experience with it, really direct experience, was in, oh, about '64, at which time I still thought that they were terribly bad, in the sense that errors keep coming up. You find out day by day that something doesn't work, or this hangs up, or that won't go. At least at that time, again, we were not supplied adequate documentation on the programs. We didn't know what to do except write a letter to the Great White Father. I often had the feeling that writing letters was tantamount to putting a letter in the wastebasket. The replies, if they ever came, were long, long in coming. So, I mean, this was when I was doing the consulting work for Northrop, as a matter of fact, and they were using the 7040s and the IBM operating system that went with it. It was really a miserable situation. Now I'm sure that since that time, the situation has gotten much better and the IBM programs, I'm told, are pretty fair nowadays.

MAPSTONE:

But in the early days you were on a different map because it would take care of your own needs.

LOWE:

Yes, definitely. Now where that concept became impractical--it was practical then but it is not practical now, and at some really fine point in between it passed from the one to the other. But I still think that if I were faced with the problem today I would still like the idea of "let's go together, throw up some money, and form a corporation and do it ourselves essentially that way. We keep control of it, but in an orderly and proper manner, not so much a bunch of guys sitting around a smoke-filled room swapping lies." [laughter] Oh, we used to--I should really say I don't know what the situation is today.

The last time I knew much about it was a good four years ago and I was removed from it; what work I've done since then has been small computers.

MAPSTONE:

Well, actually I'm really most interested in your philosophy in the early, up to the time you left Douglas, which is a key area as far, for my project.

LOWE:

While I was in there, there's a story that goes with that, too. When we ordered the 709 or the--well, 709 it was, yeah--IBM was writing an operating system for it. And they guaranteed delivery by a certain date. Meanwhile, we had written programs in the language of this system. I can't now remember its name. And we wanted to go check out these programs on extant 709s--I think it was 709, yeah, we got the first one, 7090 came later, after I left the company. 7090 was the transistorized version of the 709. To make a long story short, those programs were about a year late. Meanwhile, we had to devise our own system. And it was very, very costly and very difficult, because we had written all these programs. So we had a choice: we either wait or hope for IBM or we duplicate the work IBM has done and write the program itself or do we come up with a simpler system program and rewrite the object program that we've already written for the other system? What we did was essentially duplicate IBM's work. It was not one of my most successful projects. I had a fellow working on it who wasn't the right man for the job. It was a very unhappy experience. Cost the company a lot of money on that, lot of--tens of thousands of man-hours we poured into that thing. It finally worked and the last I heard it was still working successfully and had been as long as they had 709s and 7090s at Douglas. They never did adopt the IBM system. But that's the last I heard, and you know, that's several years out of date. I don't suppose they have 7090s there anymore.

MAPSTONE:

So going back to the early, sort of 701s in that period, you personally were writing programs, is that correct?

LOWE:

Yes.

MAPSTONE:

And did you stay writing programs or did you then become in charge of groups of people at Douglas?

LOWE:

Well, I was always the number one man in this particular installation. This again was the

computational facility of the Santa Monica Engineering Department. But I was also with programs. And I always made it my business to keep very close to the technical side of the business. Maybe I had a taste for it as much as anything else. I liked the technical side. But I had as many as 250 people working for me at the top. With a situation like that, you sort of have to decide how to allocate your time. I chose to concentrate on the technical side of the business and the customer relations side of the business. I had an extremely capable, effective assistant named Jay Featherman, and he took care of everything else: the paperwork, and the floating squawks [?]. And the last year I was there, I wrote programs. I was close to the technical side of it.

MAPSTONE:

And what were--those were Douglas, El Segundo and Douglas, Santa Monica?

LOWE:

Douglas Missiles and Douglas, Long Beach.

MAPSTONE:

And what was happening in, say, Santa Monica and El Segundo?

LOWE:

What was happening?

MAPSTONE:

Yea. What would be--Santa Monica was engineering. What were you engineering? You had your own contracts there?

LOWE:

Yes. Well, essentially Santa Monica was the commercial aircraft activity of the company. Long Beach was the Air Force. They built the big equipment, they had the big transport aircraft. And El Segundo was Navy. They built Navy aircraft. And then along in about '55 or '56--I guess it was in there--they also set up the separate missiles activity. So you had actually a split along product lines.

MAPSTONE:

Mhm. And each area had their own computing group?

LOWE:

Some engineering and some had computing, yes.

MAPSTONE:

And you were really autonomous?

LOWE:

Largely.

MAPSTONE:

And you didn't get anything [Outside sounds]--the noises! Did you, for instance, communicate on programming levels with El Segundo or Long Beach or whatever?

LOWE:

No. No, each one of these was pretty autonomous and pretty jealous of its prerogative, I think especially with regard to me as being the old-timer.

MAPSTONE:

How about the interrelationships between the various places?

LOWE:

There really wasn't too much. I, for one, felt closer to the people at the RAND Corporation, for example, than I did to the people in my own company always. We had more communication about our problems and our solutions to them.

MAPSTONE:

Oh, that's right! After RAND split off.

LOWE:

Yes.

MAPSTONE:

They still came to you for computing time or--?

LOWE:

Well, I wouldn't put it that way. They had plenty of brains of their own and so forth. But we got along very well, somehow, Paul Armer for one, and the Kerr following man, Cecil Hastings?

MAPSTONE:

Cecil Hastings?

LOWE:

We made a very nice relationship out of this ... . But, anyway, that's by the by. But within Douglas, there was no enmity by any means but I think there was jealousy. I think this was probably true to a degree in other companies, too, from what I have observed and heard. It's funny, but the partner companies just don't communicate very well. In particular, I mentioned a while ago that I worked at Northrop in '63 and '64, I guess. I was retained in part of an attempt to consolidate their several computing activities. And to make a long story short, it fell flat on its face. [laugh]. These people just weren't about to cooperate. [laugh].

MAPSTONE:

It's amazing.

LOWE:

A big company is a strange place.

MAPSTONE:

Yeah. I've got something here which I don't know what it is. It's a programming--I think it's a QUICK programming manual.

LOWE:

Sounds familiar.

MAPSTONE:

It's El Segundo, The division of Douglas, written in l954. Can you tell me anything about it?

LOWE:

Let's stop this. [Microphone off]. As a matter of fact, I think this is a little exception to the rule. I think that my operation did indeed use that QUICK to some small degree.

MAPSTONE:

Would it have been on the 701?

LOWE:

Pardon me?

MAPSTONE:

On the 701?

LOWE:

Yes. That was the machine for which it was written. Where did you get that?

MAPSTONE:

Gee, I don't remember.

LOWE:

[laugh] It's an interesting tidbit.

MAPSTONE:

These things appear. I think I might have got it from Jack Thrall, as a matter of fact, when I went through his files.

LOWE:

Could be. It would be the sort of--Jack was always social-minded, in the broad sense. He was always interested in cooperative projects and he was the one, of course, who spark plugged SHARE and this kind of partner business ... which I can't remember now.

MAPSTONE:

Guide.

LOWE:

Guide. And he might very will have actually used that, too, at Northrop on the 701—

MAPSTONE:

It's possible.

LOWE:

At North American, I should say.

MAPSTONE:

At North American. Yeah, yeah.

LOWE:

But one of the big difficulties then, as I suspect even now, with something like that, is the great difficulty of adequately documenting such a program in the sense of explaining just exactly how it would work in all possible positions, both normal and abnormal. If you don't really understand the program, and unless it's very carefully documented you can't, of course, then you're just plain stuck when something fouls up. You don't know what to do, unless you just say, "Well, I'll try to do it a different way." And this gets a little unsatisfactory, too. But all I wish to certainly say is that using somebody else's programs, and particularly something like this, is not all a one-way street. You take the bitter with the sweet.

MAPSTONE:

But that's exactly what the problem has been, hasn't it? This monstrous problem of communication. And SHARE tried to overcome it one way, yet certainly during the late fifties and sixties they didn't--hadn't succeeded.

LOWE:

It would be my judgment that they really did not succeed at all, except in spending a lot of time on it. [laugh]. Excuse me.

[Recorder off]

MAPSTONE:

I guess in sort of looking over the whole field, and I was wondering if, you know, did Douglas in any way influence IBM in what they put into their computers by sort of asking for certain things? By the time it got to the 709 and the 7090, were there things that you specifically wanted incorporated into the machine?

LOWE:

Oh, we were never great--at least, I was never great in saying what I wanted in a machine. I would say probably very little if any. Early on I had sort of formed a view--let me rephrase that. I became quite disenchanted with making suggestions to IBM, because there was never any feedback. People, little guys in blue suits, would come out with their little pink books and ask questions and make notes and then go away. That would be the last you'd hear. We would ask, you know, Joe was out here a couple

of months ago to ask me these questions, and I said, "Well, I thought about it, and not much came through as a result of it." And so it turned out that nobody knew. If you asked Joe, he would say, "Well, I'm not permitted to say. It's a classified project." Well, one of the things that led to my disenchantment with IBM really got quite, I felt, on one side of the street. Again now, this was in an era when I was one of the acknowledged leaders in the field. And so I got a lot of this people coming out to me and asking questions. So I guess the answer to your original question is "probably not much." Of course, the prize example of the alternate to that is the experience at Northrop with the prototype Card Programmed Calculator. So maybe this was my fault. But then again, I was never much of a gadgeteer, really. As I said earlier in another connection, I think of myself more as a pragmatist. I'm much more interested in getting accuracy than fiddling with the gear. [laugh].

MAPSTONE:

You wanted the tool to work so that you could get your end product, didn't you?.

LOWE:

Exactly.

MAPSTONE:

Yes. But along with your disenchantment, these tools must have had tremendous impact on the customer, in this case, the aircraft company. I was wondering of any way in which one could just look at it and say, "This was--this milestone had a tremendous impact in some way or other."

LOWE:

That's a very interesting question, and I saw answers to it while I was still at Douglas and I saw answers to it since in other connections. We can say with ease that the computer does the work of a thousand or ten thousand or a hundred thousand people and only ten people work it. Then you ask, "What would we do without it?" Well, I don't really know, but somehow I think you would simply get the work done with a lot less computation. When the computers came into wide use, specifically in the big aircraft companies in Southern California, they said "What are we going to do with all the people? They won't be needed anymore." Well, it turned out that there weren't any like that. Nobody was laid off just so they could do the [accounting] with computers. Everybody was filling out sheets for computers! [laughter] They were.

[Telephone rings. Recorder off]

I think this could be the basis for quite an interesting study in itself. I really don't know the answer to your question. Going back to the good old--where this whole conversation got started, back talking about the flutter matrices that we multiplied and tested up at

Lockheed in the forties. Nobody was laid off, and yet we were doing, I would say, probably a hundred times as much computation as had been done before. Did we make better airplanes? Well, I kind of wonder about that. The Electra, you know, killed a lot of people because of the flutter problem. And they had to ground the whole fleet of them for, I don't know how long, I forget--months, probably--until they finally found out what this flutter problem was. That's one way to look at it. Another way to look at it is that today's aircraft and other things--I'm just talking about aircraft as a "for instance," but obviously it applies to many, many other areas--today's aircraft are vastly more complex, you know, than the ones in the forties and the thirties. I suppose, if you tried to design a Phantom jet fighter without a computer, it would be something you couldn't do; you couldn't hire enough people with enough calculators to do it. I believe that's probably a fair statement. So probably--and this is a surmise on my part, and I repeat: I think it might be a very interesting study. Probably what computers have done in many areas besides aircraft is simply accelerate the technological advance. It's probably one of the main reasons we see things are changing so very rapidly now. And I suppose in other areas than technical areas perhaps, in government, in sociology, in people's attitudes, many, many things are changing very, very fast. If it weren't for computers, things just couldn't change quite that fast. Pardon me for rambling on.

MAPSTONE:

No, I asked you a very rambling question. It's one I like. It's purely philosophical, you know, everybody has their own—

LOWE:

Yes, it's purely philosophical and it's purely academic because we're not about to try to get along without computers. On the other hand, I guess I'm not really as convinced as a lot of people are that they have revolutionized our society or are likely to do so. You mentioned earlier that in ten years we are going to be far more involved with computers than we are now. And yes, of course, I agree. But I'm not really convinced that this is going to, per se, alter our society greatly. And again, that's a lot of speculation. [Chuckle].

MAPSTONE:

But there was an interesting point you made about how, with the switching from Fridens to CPC, for instance, you were able to build more planes and not necessarily at that time more accurate, that were your calculations more accurate, so you ended up with a fleet of grounded planes. So the first jump was one of just more computation being done.

LOWE:

I should correct what I feel now is a misapprehension. The Electra came out considerably later, in the sixties.

MAPSTONE:

Ah, ok.

LOWE:

After computing was well entrenched in the ... routine for many years. I just threw it into the conversation to say that despite the massive increase in computation, it still doesn't always mean you get all the answers to the flutter question.

MAPSTONE:

Right.

LOWE:

Which again is only a justification for the fact that what do you buy with all this increase in computation work that goes along with computers? That really isn't necessarily success. Maybe this flutter area is a concrete case in point. People sometimes inundate themselves with numbers that they don't know what the heck they mean, and they might be better off with a few carefully-thought out calculations on a desk calculator or a slide rule. And this is a real thing in the computational world. It's a point I fought repeatedly when I was involved in it, not just with Douglas but as a consultant later on. I try to prevail upon people to think through their programming calculations, their experimental program, whatever it is they're going to do, to plan it the way out, to describe some meaningful objectives, and then decide what is the really economical way. It is not always to just show vast quantities of data in the computer and then stand back while hundreds of papers spew out. This may be actually harmful. It may indeed--and this for sure is a speculation--be what happened in the case of the Electra and its flutter problem. Of course, a computer, if properly employed, can help solve this mass of data, and this by no means is a novel observation. That's really not a fruitful source of discussion, I guess. I still think it applies today, at least it did apply as recently as two or three years ago when I've done work for the Navy. They're computation happy. Let's just put it all in and take all the answers out, for an approach. Not a sane one, in my opinion. That's leaving aside entirely the question of waste of your machine time. But you just may slow yourself and this may just be your problem.

MAPSTONE:

Mhm. You're of the school to go the machine when you are ready, when you've analyzed your problem and done all that you can do to operate it, and then go to the machine.

LOWE:

Possibly that's too strong, but at least go to the machine with a clearly defined objective for the problem, a definite problem to be solved, not with what some of the politicos call

today "a fishing expedition."

MAPSTONE & LOWE:

[laughter].

LOWE:

All this is a generalization and like all generalizations there are exceptions, but I think there is this particular syndrome, which I would say is, "shove it all in and then take whatever comes out." People get snowed. If it's not good grammar, we'll put the [computer on ?].

MAPSTONE:

When I look back on computers, I think there are some hardware milestones, they're pretty obvious ones, but you sort of also being very much interested in software, and I wonder if you have any feelings of which were the really significant milestones in software, in programming languages and those kinds of areas.

LOWE:

Well, of course, the obvious one which springs to mind is FORTRAN, which is a tremendous advance, offered a tremendous advance in the use of computers, pretty much letting just about anybody write programs. A football player could even. [Chuckle]. We won't go into that. The--of course, I guess maybe the first milestone was simply the sort of thing we did at Douglas, getting a workable system, something that could be programmed. And let me emphasize the system, which is a significant element here, the system where you have to have, first of all, ways to get your program in the machine, ways to get your data in the machine, and ways to operate the machine, orderly ways so that you don't have to have five hundred people standing beside the machine. You have to somehow tell the operator what to do if things foul up, go wrong, if the little red light turns on and just stares and stares at you at umpteen dollars a minute. So I think probably the sort of thing that we did at Douglas and others did elsewhere with the 701 was probably a pretty significant milestone. By today's standards, it was awfully crude, but it worked, it was successful, and we did get a lot of answers. They did. I kept, the particular answer to myself. So that would be, certainly, hallmark number one. Then FORTRAN probably came along, and--and I suppose nowadays it would be simply the huge operating systems which are all based around people, which I think are still in the process of being made to work right here. And even if they are made to work successfully, why, then that will be probably about the end of the line, I think. You know, we talk about other exotic things such as creative programming, you know, a program which will write a program based on ideas. Well, maybe. But a machine is a machine is a machine, and I think it has its limitations, and so this idea, I think, is well relegated today in the area of science fiction. Again, that's the old pragmatist speaking. Maybe I'm unduly harsh on some of these bright young boys with their brilliant ideas.

MAPSTONE:

That's sort of in the future, being a media for--not so immediate. How about the problems you got into--if you got into--of bringing in, of hiring programmers in those days when there weren't programmers.

LOWE:

Well, first of all, until probably the late fifties I don't think we ever hired anybody with any experience at all, simply hired them flat. One reason for it was that there practically weren't any bod[ies]. There practically was nobody available, certainly not programmers. And most of those who were available we, well, let me scratch that. I'll rephrase that whole statement. We had, like everybody else, our own way of doing things, so if we bring in a person with programming experience he'd first have to unlearn what he'd already been taught and teach him our ways. The right way, the wrong way and the Army way. [laughter]. So, with no exception I can think of, at least to the late fifties, we hired all these bright, fresh-out-of-college graduates and taught them. And each summer, just routinely, we'd have a training course where the class of graduates, which was greater or lesser, of course, depending on the times and our needs, would turn over. Now does that answer your question?

MAPSTONE:

Yes.

LOWE:

As a matter of fact, it was quite a project in itself. We had a training manual which was quite good, with training materials, projects, and sample program. It took about two or three months for this training program.

MAPSTONE:

And would you mostly look for engineers or were there other—

LOWE:

I had my best work with mathematicians.

MAPSTONE:

Aha.

LOWE:

graduates in math. We finally settled in on that.

MAPSTONE:

Did you do your own analysis of the kinds of requirements that seemed to, were kind of background, or seemed to be best? I mean, I know IBM had a programming aptitude test, which became quite well used.

LOWE:

Yes, we tried several of these aptitude tests. We tried the one, if my memory serves correctly, that you're talking about, that IBM developed. We tried several others. And I spent personally quite a lot of time researching the matter and I gave tests to the people, after I knew what their comparative abilities were, so I could get a correlation between real life and the results of the test. Oddly enough, I found that a simple little IQ test, which Douglas routinely gave to all employees or applicants, was at least as good as anything else. [laughter]. I can almost remember the name of it, as a matter of fact; but not quite. It was a simple little two pages on both sides of the page, multiple choice question. And the correlation was excellent, which gave the best results in programming competence. So we used it. But we didn't even have to use it. Already used for us.

MAPSTONE:

The great equalizer! I have an article here by Charlie Strange, and in it he says, "Douglas designed and built several devices in the general category of computing equipment." And then I think he goes on to talk about that Douglas completed the design to computing equipment phases of the guidance systems, which I presume is missiles. But did you get into or have any involvement with what he called "general category of computing equipment development"?

LOWE:

Yes. Notably, I guess, we--I designed and had built by the Magnavox--I believe I mentioned this once before to you--oh, it came up in conversation the other day--a device to convert pdm [?] telemetry tape to digital tape. It was a fair-sized project. Of course, trivial nowadays, but quite revolutionary for its time. That was in the 701 era. And I also, something we haven't touched on, was responsible for a rather sizable analog computing installation at Douglas. We had principally REAC--Reaves Analog Computing Corporation, which included—

MAPSTONE:

Yes.

LOWE:

And I sort of had a lot of friends jimmying around with several devices that we used in that operation. But here, of course, there was far more room for the exercise of hardware imagination than in the digital area. And this was still in the days when IBM strictly forbade anyone to fiddle with their equipment. [laugh].

MAPSTONE:

And Reeves didn't seem to care?

LOWE:

Well, they sold it. They couldn't, even if they wanted to, which I'm sure they didn't. [laugh].

MAPSTONE:

What were you doing with the Reeves machine?

LOWE:

Principally plotting out missile trajectories. That was the biggest job. Beyond that, just any old differential equation problem that would come along and seemed to fit the analog better than the digital. We'd slip in the analog to do it. I don't really think of any large categories except missile trajectory type of problems. Hmm. I always had trouble selling the idea of using the analog equipment to the aircraft engineers. They typically knew very little about electronics ...

[End of Tape 1, Side 2]

[Start Tape 2, Side 1]

MAPSTONE:

Okay. This is tape two, side one of my interview with John Lowe on October l6, l973. Okay, we were talking about what you were doing with the REAC.

LOWE:

I'm trying to remember more specifics. We did some stress analysis work on it, I remember, again solving differential equations, which grew repetitious ... . It's not the only thing that it had certain uses for.

MAPSTONE:

I was just looking at this article by Sperry. Two defense calculators, five CPCs, one electrical analog made by William Miller, and REAC. Miscellaneous, 604. This is the

one that's ... 1952.

LOWE:

Well, the defense calculator, I think, is the 701.

MAPSTONE:

Right. Yeah.

LOWE:

No, not if that's '52.

MAPSTONE:

Yes, the Defense Calculator was the 701.

LOWE:

But we didn't have any in l952. May I see that? I'm curious. You asked what missiles was Douglas working on in the early fifties?

MAPSTONE:

Yes. Right. Right.

LOWE:

Honest John was one that comes to mind. Nike. Nike I. There have been two or three other Nikes since then. Beyond that I forget.

MAPSTONE:

And you were both building the missiles and doing the guidance systems for it?

LOWE:

Yes, yes.

MAPSTONE:

So that ..

LOWE:

And designing them. Designing missiles, too.

MAPSTONE:

Designing missiles, too.

LOWE:

So that when Strange talks about the computing equipment phases of guidance systems, he is probably referring to the guidance systems in these missiles.

LOWE:

Let me read it. As far as the guidance systems in missiles are concerned, I don't really think that computing, to my view of it, played any particular role. If so, I don't recall what it would be. We did get involved in a lot of things, though, involving the performance of the missiles, like the stresses and strains that developed in them due to a certain, certain maneuvers. How fast could a missile respond to a certain guidance instruction? And this comes to mind particularly in conjunction with the Nike missile which was, of course, an anti-aircraft missile. Maneuverability was an extremely important factor as to how you could correct errors as the missile approached an aircraft, and not overcorrect the errors, of course. So in that sense I suppose we may have played something of a role, but frankly this is all very hazy to me.

MAPSTONE:

Okay. So were there any others, while we're just generally on the subject, were there any other general computing equipment devices that you were involved in? You mentioned one.

LOWE:

Yes, some time later, we got E101 computers.

MAPSTONE:

The Encore?

LOWE:

No, Burroughs. I think Burroughs made it, in the early--well, let's see, early--I guess the middle fifties, I guess I'll call it. It actually was made at the old Electrodata plant in Pasadena.

MAPSTONE:

OH, yeah! Was that the 101 or was it a something-01? 301?

LOWE:

E101 is the name I recall.

MAPSTONE:

Okay, but it was the Electrodata, and before that it was Consolidated Engineering Corp.

LOWE:

As I recall, it was originally Electrodata. It was bought out by CEC and again bought by Burroughs. At the time I speak of it belonged to Burroughs. Electrodata started to develop a small drum computer. It was really quite a good machine, but they sort of turned it around and missed their market and lost their shirts. Then CEC tried to make a go of it and they gave up and passed it on to Burroughs. Of course, it's still the main Burroughs plant out there. Burroughs, I guess, has been doing all right with it. But they produced this little desk size computing machine which was strictly for, well, we set it up for just local shop use. Anybody who wanted one, wanted to use one--we had several scattered around the department. It was really what I suppose you might call Douglas' first adventure into open-shop type of computing. It was a form of a general purpose computer. But it had pin boards by which you could program the machine and all data entry was through a keyboard.

MAPSTONE:

Was the pin board something like a plug board?

LOWE:

Something like it, but not quite. You just stuck a pin in and that closed the connection down inside the machine. But it wasn't a matter of point to point connection, as in a true plug board. Let's see, I'm really having to fish hard in order to visualize this thing. Anyhow, it was quite successful. It was a good machine. It was reliable. We decided we could add greatly to its usefulness if we could feed information to it through punched cards. So in our own little shop there, I had a couple of devices to--you read the cards on a keypunch machine and fed the information into the Burroughs machine, so you could make up your program, put your data on the cards, and feed it through the key processor. They would clang, clang, clang narrowly through a big awkward box about three or four feet long and a couple of feet high. We built it carefully for reliability and I never heard of one of those boxes giving us any trouble. We selected the most reliable relays we could find. [Chuckle]. Then there was another project. It was kind of interesting, although engrossing, in fact, not very onerous. I took the device, I remember, into the analog room for recycling a program to get a new parameter. The typical procedure with an analog problem was to have the operator sit there and run a solution, twist a knob and

run another solution, to develop a family of curves. And we hooked up something ... we needed to recycle, and then it would pick up new parameters and start reading all by itself. No use going into details, which are pretty hazy now, too. Also developed a scheme for putting functions into the machine through magnetic tape, recorded at an appropriate level representing some arbitrary function on magnetic tape. And you'd read that into the machine instead of using the other available means for putting the functions in the machine. It was sort of in the same ballpark, something that would enable a function to be recorded on tape and then read back into the machine at a certain time later, functionally reducing a t_1 minus t_2 function into a differential equation. I really twisted IBM's tail and made them mad when we first got the 709. I wanted a clock for it so that a program could reach out and find out what time it was. So IBM [wouldn't], so I got one and hung it on the machine.

MAPSTONE & LOWE:

[laughter]

LOWE:

And it was big! Everything I ever built was big. It may not be much for pretty, but it's hell for stopping! I carefully devised it so it covered the IBM logo on the console and I put the Douglas logo on at the top, so the whole thing looked like a Douglas computer. It was quite a clock. It kept good time and we could, by appropriately good command, design a program to find out what time it was to a thousandth of a second. [laughter]. Including the year, by the way. The guy that wrote the program to read the clock even built in leap years!

MAPSTONE & LOWE:

[laughter].

MAPSTONE:

That's crazy.

LOWE:

Oh, that was fun.

MAPSTONE:

So you talked a little bit about having a closed shop, which made me think about just that subject, which seems to have been an ever-ongoing battle. I take it you were originally a closed shop.

LOWE:

Yes. ...

MAPSTONE:

What led to your becoming or thinking about becoming open?

LOWE:

It just seemed to me an actual revolutionary thing. Some of the people in Douglas, we worked in the computing section, didn't make us the only people interested in computing and the only people who could handle programming and do computing jobs. So this little E101 thing was our first adventure, and this branched out then later and we got some more sophisticated, more powerful machines: the Librascope LGP30

MAPSTONE:

Oh.

LOWE:

and a couple of others, at which time I sort of dropped out of it. I ... as you would have expected, two or three people in the department had displayed a particular interest in this kind of thing. So fine, I just, you know, let them carry their own ball, and I didn't pay much attention to it. I don't think there was any bad feeling or anything, just--and they indeed came around with questions on some matters, [but] they would operate them all by themselves. The problem is, where do you draw the line on something like this? If you let the thing go out of control, then pretty soon every engineer has his own 360. Of course, with consoles and the modern techniques, I guess to a large degree you can do this. I realized a considerable danger if you regard your costs at all, in that you can sure waste a lot of time on a machine. And modern machines are not cheap. So there's the whole thing in a nutshell, as I see it at least. How much control should the professional computing people in a certain organization exercise over the use of equipment? How much efficiency can their supposed "expertise" add to the program? When is it better to have the professional, when is it better to have the guy do it himself? I think there's no iron-bound rule to answer this question. Among other things, it obviously depends heavily upon the people involved in it, as well as the company and the situation involved. Otherwise, I guess I don't really know the answer to that question. [laugh].

MAPSTONE:

So you were running both then in the latter part of the fifties?

LOWE:

Yes.

MAPSTONE:

When did you get the Burroughs machine?

LOWE:

Gee, I'll have to guess about '58. I can't think of a single piece of documentation. But I'll tell you where you could find out. There's another old-timer, by the way [cough], who has now recently, I understand, gone back to Burroughs. He was at Burroughs when we were getting those E101s from them, and I remember I first got acquainted with him. Oh, gosh.

[End of Interview]