An Interview with

CALVIN N. & CHARLOTTE D. MOOERS

OH 254

Conducted by Kevin D. Corbitt

on

22 June 1993

Minneapolis, MN

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

Calvin N. & Charlotte D. Mooers Interview
22 June 1993

Abstract

Calvin and Charlotte Mooers discuss their association with information retrieval and programming language research from World War II through the early 1990s. Calvin Mooers describes his work in the Naval Ordnance Laboratory (NOL) computer project and his decision to leave NOL for graduate school at MIT. He discusses his development of Zatocoding, an information retrieval system, after discussions with J. W. Perry in 1948. Mooers recalls the formation of the Zator Company in 1949 and his attempts to patent and market his system. He discusses how his frustration in the field of information retrieval led him to explore other interests and the shift in Zator's focus to research with the addition of Raymond J. Solomonoff in 1957. Mooers discusses the research advantages gained through the organization of the Rockford Research Institute, Inc. He describes Solomonoff's continued research in inductive inference (artificial intelligence), his development and attempts to market the TRAC programming language, and his work with Eugene Stuart Fergusson on ASCII standards. Mooers discusses his involvement with Data Concepts in the late 1970s to use TRAC to develop a software package that printed insurance policies. He recalls the company's dissolution in 1983 although the software, SIMPL!E, had been completed. Mooers describes his redefinition of TRAC into TRAC-2 and attempts to market TRAC-2 following his work with Data Concepts. The Mooers conclude the interview with a brief discussion of his current projects related to TRAC and writing vignettes on the history of computing.

CALVIN N. & CHARLOTTE D. MOOERS INTERVIEW

DATE: 22 June 1993 INTERVIEWER: Kevin D. Corbitt

LOCATION: Minneapolis, MN

CORBITT: Could you begin by giving a little bit of biographical information about yourself? I know you were raised here in Minneapolis.

MOOERS: I was born in Minneapolis October 24, 1919. I went to the public schools. The first school was Logan School and the intermediate was Franklin Junior High School and then North High School. I wasn't the best student, but I was in the upper part of the class and kids and teachers pointed me out as being "the brains." In junior high school and high school I started collecting very good grades -- mainly As and some Bs. For that reason in part, the last year of high school they put me ahead. I was in high school only 3-1/2 years and it balled up the alumni records so I got formally put into the 1937 June graduating class instead of the 1938 January class. That is material because of the 55th reunion which I just went to, which was a very pleasant time.

My mathematics teacher, Viola [A.] Marti of North High School, had taken some summer math teacher courses at the University of Minnesota. In connection with having taken those courses, she called up some of her teachers and told them that she had this student who would be interested in talking to them. That is, she worked up an introduction for me to the University of Minnesota. The upshot of that was that I got started early in the University of Minnesota, and part of the reason for changing of the graduation date at the high school was due to this introduction. The University said, "We can let Calvin come in in September with advanced placement, and during the summer before, he can study and take some advanced placement exams so he can start right out." So the first year at the University I started in mathematics with calculus and analytic geometry courses. The placement exams were in algebra and trigonometry, and evidently I did satisfactory in them so they gave me advanced standing, with the result that in the second year at the University I was essentially taking senior courses and beginning to take graduate school courses. In other words, in the second year I was taking courses such as differential geometry and, well of course, I got the usual courses in the humanities. I took a course in economics and the like. I took several logic and

philosophy courses, also physics. However, my eagerness and application seemed to continue so that while at the University I got also mostly straight As. Minnesota, at that time, and maybe now also, had a provision that you can translate very good grades into essentially credit equivalents. So after three years I was one credit short of the requirement for graduation. At the beginning of the fourth year at the University, I was formally in the graduate school and taking what they considered their toughest graduate courses in mathematics and physics.

The fourth year was September 1940 going into 1941, and in the Spring of 1941 the European situation was getting more and more complicated. Several of the University professors had been called to Washington to organize various laboratories and one of them was Professor Lynn [H.] Rumbaugh of the University of Minnesota. Rumbaugh taught my graduate course in theoretical physics. I took that in my third year as an undergraduate, and so I had completed theoretical physics, being one of his better students in the physics class.

Now we get into the spring of 1941. Rumbaugh came back to the University and was recruiting students to come to Washington for a laboratory that he was helping to organize, as they were expanding some of the work in Washington. Rumbaugh recruited me to come to Washington as a contract employee in the Naval Ordnance Laboratory late in the spring of 1941, and remember, Pearl Harbor was December 7, 1941. So I was not here to go through the graduation ceremony in 1941. I left Minneapolis and I've been away from Minneapolis ever since except for visits.

For graduate students recruited in this fashion they had an arrangement -- a contract employment. In other words, bypass the red tape of civil service. So I became a contract employee at what my father considered an outrageously exorbitant rate of return. It was a dollar an hour, eight dollars a day, flat. We started out five days a week and then later on when things got even heavier in Europe we were working six days a week. We worked all day Saturday, but we got out an hour earlier, or something like that.

CHARLOTTE MOOERS: Well at that time we were working nine hours a day rather than eight, and on Saturday we worked only eight [laughs]. Fifty-six hours a week.

MOOERS: I was recruited as a physicist and my first work was antimagnetic mine measures. Another part of my background was that I early on became very interested in the radio. Actually my father and my grandfather in the early 1920s had bought radio sets -- Atwater Kent -- the units were wired together and so on, made tuners and crystal sets and all that sort of thing. Also my grandfather had a hearing aid which was a box about this big with a handle on top and ear phones and a large battery [laughs]. It worked, but not very well. Neither does this work very well [indicates his own hearing aid]. However, so I got interested in radio, and by the time I was 13 I had built some sets and met some other boys. The great temptation of course was to make a bootleg radio station. The boys were doing that and, I'm not sure, I probably dabbled in it, but I actually went and learned the codes well enough to pass the license so by the time I was 14 I had an amateur radio license, and of course that took the fun out of being a transmitter [laughs]. But I did have a license, and I did build some receivers. This gave me quite an intellectual and emotional background in working with electronics of the day, and with electricity, and so I had great fun studying about radio. *Popular Mechanics*, radio circuits and schematic diagrams -- that is, picture diagrams rather than the technical circuits -- and soon I learned to read circuits themselves and bought the components and made a receiver and I did actually make a few contacts by radio in code.

This was relevant because it gave me a very personal detailed background in putting electronics together which was used in the laboratory. I was hired as a physicist and later on I actually built some acoustic devices for acoustic measurement and so on, and from radio I knew what to do. Another thing that I was very interested in was finding why these things work. What was a radio wave? Why did a radio wave wave? And it became clear in order to do that, I had to have a better understanding of mathematics. Mathematics was the descriptive language of radio at all levels. Theoretical, that is, the physics of radio waves, the physics of acoustic circuits, oscillator circuits, and so on, the physics of the antennas. And here I was, I started out with only ninth grade mathematics. Eighth grade, of course, is a wreckage devoted to long division. Only in ninth grade do they let you on to the fact that you can do algebra with Xs and Ys and unknowns and solve the equations, which was a great insight because it increased the power. I actually made a very conscious resolution to set aside my total focus on amateur radio and to move my primary focus to study mathematics early in high school. The major decision on this probably was forming in junior

high school, that is in ninth grade. The senior high school was 10, 11, and 12th grades. So by the time I was in high school and taking the courses there, I was definitely wanting to learn all I could about mathematics. Well, it turned out that the people in North High School -- a typical American high school -- knew very little about mathematics. They had been trained by rote to teach algebra, trained by rote to teach trigonometry, and it even went into the people in physics. I still have at home in my desk or at my computer a slide rule which I bought when I was in high school. I got bawled out for using this slide rule in high school physics to perform experiments. Mr. Houston, the teacher, insisted that in physics experiments, all the numbers must be computed out to two decimal places -- two decimal places, no matter what the magnitude of the number was. Well, of course, with the slide rule, you become aware that this is nonsense. But this orientation, and the interest in aspects of mathematics which couldn't be answered by high school, were what Miss Marti learned about and supported me in, and got me the introduction.

These are, as it were, early history. The fascinating thing was that while I was still at North High School on several occasions, as a result of the introductions, I took the street car over to the University here, and in Folwell Hall went to the several of the weekly tea parties of the mathematics department where they had the weekly seminar of the graduate students giving their talks. There I was, a high school student, welcomed to these sessions. It was indeed pleasant and reinforcing. I met the professors who I later took courses from, and who from that time on knew me. In their knowledge of what I was doing, they persuaded me to get into the liberal arts school rather than the engineering at the University. They said their reason was that mathematics was better. So this meant that I had an entrée as a way of introduction to all of the professors before I got to the University, and it was emotionally very supportive. It made me more desirous than ever to get good grades.

So that was the background, and then Rumbaugh enticed me and several of the others to Washington. It was from the stay at Washington that I got into the computer work. Now the computer work started in 1945, just after that point where the acute war hostilities had ended. At the laboratory they were aware that they had to figure out what we were going to do now that the war is all over. What do we do with these people? What reorganizations are appropriate for a Navy laboratory? Also [John] von Neumann, who was traveling around as an inner-circle advisor to the Navy, had wanted a computer for himself. He thought that maybe a good way to get a computer would be to

talk the Navy into building a computer, so that's what he did. Well, he talked the Navy big wheels into it, and they searched around and said, "Well, where should we put a project like that? What about the Naval Ordnance Laboratory where we have these people that were working on electronics for acoustic mine measurements. They have a lot of bright people. They have a new laboratory out in White Oak, Maryland just outside of Washington." And this new White Oak Laboratory was just built. The buildings weren't occupied. Many of the buildings weren't completely finished, but one of the buildings was finished -- an all-wood building which was built for magnetic measurements. Well, it turns out by the time they got the building finished, the Germans were into acoustic mines and pressure mines and things like that which were not based upon magnetics because the antimagnetic mine methods were getting successful enough so that although they could still use them and blow up a ship, it wasn't as easy as it was at the beginning. So they were going into different mines that were easier. So there the new building was. It was larger than a cottage, but it wasn't a steel and concrete building like the usual laboratories. The main room at ground level was about this long and a little bit wider [indicates a room somewhat larger than small conference room], and over the office side of the room there was a platform floor up above. The roof was coming down on one side, and a fence of wood railing on the other. This was where I had my desk. Well, the thing was that the people and the computer project that was formed landed in this building. Without getting into the details of it, J[ohn] V[incent] Atanasoff was in charge. I was up on this upper level, and you [Charlotte] had a desk down on the floor level.

CHARLOTTE MOOERS: The Acoustic Division was down there.

MOOERS: Atanasoff was running these two divisions (Computer and Acoustic) which was presumably against all rules, but people were leaving and he was staying. Nobody really wanted to head the Computer Division. What good's a computer [laughs]? It's just a wild idea. So it was tossed around like hotcakes. First G. H. [George Hiram] Shortley, who's name is very famous in quantum mechanics, was the temporary head. My transition was made through a group headed by one of the Slawsky brothers. There were two memorable Slawsky brothers at NOL: Mitch and Zaka, identical twins, short and stocky. Zaka was briefly my boss. Then Atanasoff took over. There was a very brief period when the computer group existed only on paper and with only one or two people in southeast

Washington, the Navy yard there, before they moved out to White Oak. But with the move out to White Oak, they started putting it together. They started recruiting people and transferring other people from the laboratory. At White Oak the computer activity went on from sometime in May or so of 1945, and the project terminated after I left, presumably some time in December of 1946.

I left in September of 1946 and the reason for it is a story I have. Atanasoff, from what he said and what has been written (I haven't talked to him about it), says he doesn't know the reason why the project was terminated. Well, of course, under the circumstances when the project was a failure, you don't tell the wheel that was in charge of it exactly why you terminated the project, not unless you were ready to handle the consequences of doing so. So the decision came from the top that it was terminated, without giving reasons. So Atanasoff did not take any blame. In retrospect, of course, it seems to be from what -- I haven't researched the project -- people tell me that Atanasoff thereafter claimed that all we were doing were feasibility studies for a computer. Of course nothing has to be successful if it's feasibility studies. Well, it's true that's about as far as we got. We didn't even get any good feasibility studies, but we didn't get anything primarily because of him. Atanasoff was the reason that nothing happened. I think I can say that flat off. Of course what I wrote up for the *Annals of the History of Computing* (volume 15: no. 2, pages 54-55) says essentially the same thing.

All I can do in a little account is to give instances of what was going on. I was about 26 at this time. Draw a comparison. [To the interviewer] Given the situation, let's say, that you have an immediate very senior boss who was cooking up some wild idea which was against everything that seemed to be reasonable. Would you then be inclined to find your way to the head of the libraries or the president of the University to shoot him down and complain about what's going on? Would you be sufficiently self-confident that you knew the facts so that you could reasonably complain that this is not the way to do it? Well, that was my situation at NOL, and so I was trying to hold the project together and he was shooting it down by his inaction, but shooting it down with a smile at every turn. We were just not getting anywhere at NOL in our computer project.

I wanted to go to graduate school. I got accepted for the graduate program at MIT, but, here with this background in

computing, I wanted to do something with it. So in my own mind I set up a list of altematives. With this unbelievably unusual education in something that was going to be very important -- which was computers -- what could be done with it? Well, I turned over the list of things. One possibility I thought of was ethnic language translation. Others were pattern recognition, that is of graphic patterns, and dealing with them. Another was what I call thinking machines.

TAPE 1/SIDE 2

MOOERS: The fourth and final area needing attention was library activity. You see, at that time we had all been writing scientific reports in all of the laboratories, and so the government was just awash in reports which presumably had very valuable information in them. They were classified for security purposes which proves they were valuable [laughs]. Yet the techniques of library classification were clearly very primitive. So that was the fourth one on my list. After considering the list, the upshot was that I decided that most of these probably would require a working computer, and there were no working computers. While at the Navy Ordnance Laboratory I was a volunteer in a little project to try to try to figure out how to rationalize some of the operations in the library report classification, so I had gotten into that field just a little bit and so that probably tilted my interest. So I decided to see what could be done with what I knew about digital processes and mathematics in improving the situation in libraries. So that's the alternative I chose. Not long after (at MIT), I went to a lecture by Claude Shannon that he gave about "information theory." One of the conclusions of the lecture was that a random process had the statistics required for passing the highest quantity of information. Of course, a random process ordinarily is pure noise. Well, the whole message of information theory turns on the matter of choice, and choice in comparison to other choices. So that lecture stuck in the back of my mind and impinged on some other ideas that I was speculating on. This got me into the field of "information retrieval." In other words, how to classify and recover information in a systematic way.

What makes it a good story is that my interest in the idea of randomness resulted in an invention (Zatocoding) for which I applied for a patent which was eventually issued 23 years later. I interested my brother, Howard [T.] Mooers, to come to Cambridge to see if we could commercially do something with the idea. What turned out was that this

interest spoiled my habits of getting the kind of perfect marks that I had gotten at the University as an undergraduate. The folks in the math department at MIT just weren't interested then in what is now known as "information science." To make a long story short, they eased me out of a doctorate program at MIT because of my premature interest in information science. So I wrote a masters thesis involving some of what I was doing, with the process that I had invented, which was Zatocoding, and my Zatocoding patent was issued 23 years later. Clearly, the business opportunity had disappeared completely by the time the patent was issued. That got me into my post-NOL record, which is the archives collection you have here [at CBI].

It would possibly be most useful for me to set out a framework for these reminiscences. I will just sketch out from the time I left NOL and go on. In 1946 I came to MIT in graduate school and was there for a year and a half. I graduated with a masters degree in mathematics with a paper on statistical use of Zatocoding for analyzing records, which is not the actual title of the paper, but that was the content of my masters thesis. I came to MIT in the fall of 1946 and by 1947 I was very interested in the field which is now called "information retrieval" -- in other words, library work. At MIT there was a person on a sort of back door arrangement, with an office up on the fourth floor of building 2 or building 4. His name was James W. Perry, a chemist who was fascinated by the Russian language. He was writing a book, an introduction to Russian for chemists. He was also interested in the problem of what was called then "chemical literature." This concerned methods of indexing and finding information in the field of chemistry. The American Chemical Society then had a chemical literature division which had a newsletter, I think, and also sponsored section meetings at the annual national conferences. Perry succeeded in getting me put on the Natonal program for one of these meetings in which I described some of my ideas that I had at that time (this was prior to Zatocoding). The idea was for a scanning mechanism in which mechanical records on film would pass through photo cells and be recognized, and references selected and put out. In connection with my description I gave, perhaps for the first time it appeared in the literature, a description of the use of the Boolean operatives of "and," "or," and "not," for describing a library selection. This was in September of 1947, and I promoted the idea that it would be a great thing if the chemists were involved somehow to making such a machine. Of course it was visionary then, but it stirred up thinking. At that time the chemists were playing with the use of index cards with marginal holes and notches for sorting cards. This is an interesting system, and J. W. Perry was fascinated with it. He was one of the

organizers of these sessions. They would have chemists come in and talk about their punch card systems. The punch cards would be either a card with edge notches to perform the sorting, or would be the Hollorith-type (which are now called IBM cards) which could be sorted by running through a machine. Perry was interested in advancing interest in that sort of thing. Somehow we got in touch and we spent a good deal of time together talking about things. This relationship, and this interest by the chemical society, became, you might say, the fertile field for development of the Zatocoding invention which I made in 1947.

Without getting into the technology of it, I became convinced of the idea that this was of value to the library activities epitomized by the technical librarians. Not so much in a library like the Walter Library here [at the University of Minnesota] where you have a vast collection of books, but presumably the libraries with more recent publications which were coming up on just all topics. The literature of the wartime scientific laboratories did not fit into any of the library classification systems. That's what was disturbing people and confusing them. Here was an inundation of new material, but how do you deal with it? So Perry was trying to devise methods, and a theory and an approach, and to interest people in it, and I was the inventor looking for solutions. So I did develop a solution. I got my brother, Howard, to join me in Cambridge and we set up the Zator Company. We actually produced a sorting mechanism, which you've seen the pictures of (in my collection at CBI), and the Zatocoding technology. Over the years I did succeed in getting some 12 commercial customers that did pay in various ways for using the Zatocoding system. It would be a very interesting story of market dynamics and people dynamics to go through this history. However, after a year and a half it was clear that it wasn't going to be any sort of overnight success. It was costing me and my brother Howard money to support the activity, so we amicably separated. He came back to Minneapolis and got into work with Honeywell. The field of interest that he got into was transistors. Incidently, [John] Bardeen, the inventor of the transistor was one of my professors at MIT. Charlotte had Bardeen as a boss or supervisor in some of her computation work at NOL, and I remember visiting the Bardeen residence as a student here at the University of Minnesota for, I think, a New Years' party at his house.

CHARLOTTE MOOERS: Did you make clear that Bardeen was at the Naval Ordnance Laboratory?

MOOERS: No. Bardeen was another one of the people who was enticed finally to come to the Naval Ordnance Laboratory, but he never did get into the computer activity. He was busy with other things at the time the computer activity started, and was not interested. Bardeen left for, Bell Labs, and the University of Indiana where he more or less stayed.

Well anyway, getting back to my own story -- about the Zator Company, the Zatocoding System, and its sorting device. My brother left. I continued more or less single-handed seeing if I could market the thing. I had applied for a patent and it was a high camp intellectual burlesque in the way the patent office responded to my patent application. You should read there objections! Twenty-three years later I got the patent. The patent office wrecked the commercial potential so far as I was concerned. But I had set up Zator Company and I had established an office outside of Harvard Square in a building. Initially I used one of the floors. It was an old residence. It was attic, second floor, first floor, basement. The basement was for a furnace and the attic was storage. I used the second floor. I rented out the first floor for a certain period. We had an architect, and then another fellow working in public relations, and so on at different times. So I had an office there, and I held forth and did a lot of traveling hither and yon trying to market the Zatocoding system. Now it was a complete system. In terms of contemporary lingo, I think it is useful to describe it. It was non-electronic, which equals mechanical; it was digital; it would be called a knowledge-based system; it was a selector device (you didn't pick the cards out by hand); it was automatic, in other words, a motor drove it; and it exploited the Zatocoding technique which can be characterized as selection based upon "fuzzy sets." This was a fully formed and operating system, say in January of 1948, without an electronic computer. Well, I state it this way because if you describe it as I have, it shows the relationships. The first commercial sale was in the spring of 1948, to Merck, Sharp and Dohme of Philadelphia.

So I continued to travel and try to market this and gave papers, and people found it interesting. Most of them were buffaloed by it. They couldn't really understand it. It antagonized librarians. So why did it antagonize librarians? I was describing an operation which, in the field of library science, live human librarians were considered to be required. This was contrary to their vitalistic idea. They believed you had to have a living human brain to perform such a selection. Yet, I was trying to demonstrate that you could do it mechanically. I did it, and it worked!

However, this was repugnant to the librarians. Some of them took offense against the idea even though they weren't able to fully formulate why they were offended by it [laughs]. This was what occurred. As you put together the historic record you would find I was doing quite a bit of traveling, and I began to be invited under different auspices as a consultant. I was invited to Norway to give a special course in information retrieval. I was asked to go overseas to Germany and they paid part of the cost to put me up to attend a meeting they had there on information science. This was Erich Pietsch in Germany. We developed quite a bit of a friendship. So I was in Germany when the Cuban missile crisis occurred. I was not only in Germany, but the meeting was at a spa outside of Frankfurt. So here we were doing wine tasting and so on while back at home they were having the Cuban missile crisis. I wasn't able to speak German or to read the newspapers, and they were embarrassed trying to explain what was going on. They couldn't understand it either, but here was this world shaking event taking place.

One of the major consequences of that sequence of my activities was to force into the consciousness of people in library science and computer science that the two fields had a common area although it wasn't fully understood or defined. I was creating a commotion with the imputation that here was something important that these new digital methods held for library science, and that the Zatocoding system was a case in point. My invitations to these meetings and the willingness of other groups in the U.S. to have me come and submit papers indicated that this field of turmoil and prediction was developing status. I think it is fair to say that I had played a very significant, but not a sole portion of that development. My mathematics proclivities and background gave me sort of status that James Perry did not have since he was a chemist and he was cautious about making any statement that was quantitative, mathematical, or theoretical. I was willing to come flat out and use the word "theory" and to say that mathematical science had a bearing upon what they were trying to do, and making machines for this, and for saying that machines are the appropriate tool, and machines are going to come.

Then somewhat later, Mortimer Taube appeared as a late-comer. What he did was that he cooked up a simplified variety of my descriptors which he called Uniterms. He was a great salesman and a smooth talker and he charmed the librarians. He had worked as a librarian. So he set up Documentation, Inc. which made quite a commercial splash. Taube's message was that you don't have to worry about the fact that you can't understand Mooers, you do it the

Uniterm way, you can understand it, and it's easy. So they flocked in his direction. Well, his methodology can be cynically characterized as follows: How do you index documents? You take a collection of documents in a certain field and you give them to somebody that is not really in that field. You sit him down with a colored pencil and ask him to go through the documents and to underline every term that he doesn't understand [laughs], and to use those underlined terms for index terms. You've heard of key terms, key words? Well, key words are the direct descendants of Mortimer Taube's Uniterms and have the same sort of loose-jointed semi-applicability to the field at hand.

Well, marking out this history, I'm trying to go through the skeleton. I got an associate, Ray[mond J.] Solomonoff, who was at the historic Dartmouth meeting of the artificial intelligence people in 1956 or 1957. He was one of the invitees along with Marv[in] Minsky and John McCarthy and others. I was not there. Whether they had Newell and Simon, I don't know. They had a meeting about artificial intelligence. Ray has called his work a theory of inductive inference; McCarthy called it artificial intelligence; and I called it thinking machines. The thing was that this was his field. Beginning in high school, I was terribly fascinated by finding people who were extra bright and dealing with them in one way or another. Well, Ray Solomonoff was very bright and had these wild ideas, and so I asked him to join me in Zator Company, and moreover we put together a proposition. I got him to write up a research proposal to the government. I had not done any research proposals, but I saw what was developing in the field and thought I'd try it out with Ray.

TAPE 2/SIDE 1

MOOERS: He and I put together a research proposal and submitted it to the Air Force Office of Scientific Research, to Harold [C.] Wooster. Wooster was mystified and charmed by Ray [laughs]. Here was this guy with these wild ideas, an ethereal sort of intellectual. Later on he started wearing full whiskers which were then black and now they're white. But anyway, in 1957 we had perhaps the first government research grant ever awarded for artificial intelligence work. You see, McCarthy's work was supported within the broad umbrella of some university's broad scale research program. Here was a de facto research proposal for artificial intelligence awarded in 1957 to Ray Solomonoff. We continued this relationship on a year-by-year basis with Harold Wooster for four or five years. This started Zator

Company in the business of setting up proposals for contract research. Zator Company was a proprietorship. It turned out later that you could not give a contract to a proprietorship because it was a company owned by people. If, however, you had a non-profit organization, where there was no ownership, you could get research grants which were much freer of control. So for this reason, I organized the Rockford Research Institute somewhere about 1960 with the purpose of continuing research work in information science and artificial intelligence. We put together research proposals which related to information science. Also Ray continued his work year after year in inductive inference. He was discovered by the Russians and began to be much more cited in the Russian literature than in the American literature. So, over the past five years he's had contact with some of the Russian people, and he's still doing work in inductive inference. It's gone beyond me for my actually being able to understand the research, but the fact that people want to publish his papers indicates that it has merits. So that's still going on. I might say that with the formation of Rockford Research, Zator Company became number 2 in the operation and Rockford Research became number 1.

CORBITT: Rockford Research Institute and Zator were at one point contemporaneous?

MOOERS: Yes, and at the same address. Rockford Research was a non-profit corporation. What that means is that it was licensed as a corporation under the laws of the state and it has a set of trustees rather than owners. Non-profit is a total misnomer. I made more money under the non-profit corporation [laughs]. You can even make money on research grants. Of course, what you do is to pay it out in salaries or in travel money to use. It gave a great deal of freedom and it was running contemporaneously with Zator Company. In the early 1960s, all of the grants and stuff were transferred over and were reissued as they came due under Rockford Research. So I think under Rockford Research, we had support from Harold Wooster for Ray Solomonoff. From the National Institutes of Health we had support for myself and Ray and then E[ugene] Stuart Fergusson, who was put on as a half time employee. He had his own business he was trying to create. He was consulting in the field of communications. He had been in the Navy and became one of the junior officers. In the Navy he was working in the field of radio communications including telegraphic digital communications. Then a government organization snapped him up when he got out of the Navy and he continued working in that field. It was while he was there that he fell in with Arnold Cohen and the guy that

later created Control Data, William Norris. At this organization Fergusson was used in supervising extramural contracts in the field of telegraphic communications and so on. He became the hatchet man, as it were, to police some of the faulty contracts, to terminate them, or to go to conferences and sway the conference in such a manner that suited the purpose of his boss. Norris was not really in the organization. There was a precursor of Control Data which was set up to insulate it from the agency and it was given another name, one of these forgettable names, [Engineering Research Associates]. Norris was the wheel there and Norris wanted Fergusson to join him too in the project. Soon thereafter, Norris pulled up all stakes and took, I think, Cohen along with him and formed Control Data. Now it turned out that the path of this interim organization and my activities crossed. I never met Fergusson, but in the NOL computer project I got hold of some iconoscope tubes and I got hold of them through this company [Engineering Research Associates] [laughs].

Well, anyway, Fergusson joined Rockford, and along the line at the beginning of the 1960s when I was becoming interested in TRAC language. I was becoming more and more critical of what I could do in the library field. That is, by 1960 there were now computers and "operators" like Herb[ert R. J.] Grosch at General Electric (GE) were moving in, and being the big boss of the computer at a company and were going around looking for business. And the library field was beginning to wake up to the fact [that] there might be something here. You don't take your business to a little hole in the wall like Mooers was operating. You take it to GE or you take it to MIT. There was an "operator" -- Overhage at MIT -- who set up a big project, INTREX, to solve all the problems for all time of libraries with computers at MIT. Herb Grosch was taking contracts at GE. This was the situation. The result of all of this was that in the mid 1960s, I more or less turned off my public interest to the information and library field, although I kept following it to some extent in private, and turned on my interests in programming languages and TRAC. I developed TRAC between 1959 and 1964 when I finally defined it. We got a contract from the ARPA to see what we could do with TRAC to get it programmed, and with the money from ARPA we did several things.

Fergusson and I started to go to standards meetings with ARPA money and blessings. One of the results of these meetings was that upon my initiative, and with Fergusson's connivance and help, we set up a petition campaign by academic institutions directed to the ANSI standards organization. It was to the affect that "You shall not let the next

standard come out for an ASCII code which was monocase." You had to have upper and lower case. Well, this was totally upsetting to the large companies and to IBM who wanted monocase only. However, we brought it out in the open. We made this petition, and it became impossible to shoot it down. We also made the imputation that IBM was manipulating things in order to maintain their degree of monopoly in the field. You see, if you could restrict it to monocase you would have one kind of engineering computer and, if word processing ever came into being, you'd have the second kind of computer, and never the twain would meet because of the need for separate codes. Well, we shot that down and I'm quite proud of it. I think that was one of my great accomplishments.

In 1965, I hired L. Peter Duetsch, who was a high school student 17 years old, who did the first programming of TRAC. He got TRAC running in the PDP-1 computer at Bolt, Beranek & Newman. Not long thereafter we had a telegraph wire, that is a DC telegraph wire running between BBN and my house and also my office in Cambridge, something like four miles total round trip -- a triangle. It took 250 volts to push 60 mils through the wire [laughs]. At either my home or at my office, when I went to my model 33 teletype and turned on the motor, I was online with the PDP-1 at BBN. This was 1965. I can honestly say that for the next 15 years my computer service went downhill.

I wrote up a descriptive account of TRAC in 1965 for the *Communications of the Association for Computing Machinery*. The editors, when it came in, evidently were amazed since they had never heard of me. Of course they hadn't heard of me. I hadn't talked about programming languages. I wasn't one of the big "names." So I came in with this finished piece of work, and they sent two of their big wheels out to look me over. One of them was Carlos Christensen and the other was Robert Floyd, a big wheel in parsing programming languages. They came to visit me at the office to find out who this guy Mooers was and how come they'd never heard of him. I brought them into the office and took them into the back room and turned on the teletype and we were in remote communications -- with a remote computer at BBN on which TRAC was running and I demonstrated it. So they were all ready to deflate a hoax [laughs]! Quite different was the fact of the matter. So my paper was published.

Then I tried to promote TRAC as a commercial product, but there were difficulties. One was that in those days computer software was not considered property. There were no rights in computer software, the ideas were

"foreign." There was the belief in the legal profession that neither copyright nor patent would apply. Of course, then in the 1960s, I still hadn't been issued a patent on my Zatocoding. What I was interested in doing was, in one way or another, to make an economic capability out of TRAC, which was a clever creation. It still is. What happened in fact was that TRAC probably became the most widely bootlegged computer intellectual property that existed. In other words it was terribly easy to program and it was programmed at one place after another. For instance, Dartmouth had a version called CART -- TRAC intentionally spelled backwards. Professors assigned programming TRAC as a project. And there were all sorts of other implementations called by various things including TRAC. I was trying to market it, and since I could not use copyright nor patent, I was trying to use trademark. The first issue of *Dr. Dobb's Journal*, one of the early publications in the personal computer field, has a vitriolic editorial against Mooers and his rapacity in trying to charge people for his computing language [laughs]. It makes interesting nostalgic reading, and you can look back now and can see that Bill Gates has created one of our biggest economic enterprises [Microsoft] on the basis of this intangible we call software. Of course he's had a lot of luck. A very skillful, lucky, fortunate, course of events that he's administered. So during the 1960s I was trying to do something with TRAC, and also was investigating what it could do without any real organization behind me.

Then beginning in the late 1970s, two individuals in the neighborhood of Boston, Thorpe E. Wright and Andy P. Diamond, were putting together a little company. Thorpe Wright was a consultant at Arthur D. Little and Andy Diamond was a salesman at Nixdorf Computer. Andy had been putting together little software packages to assist people in the insurance industry. The package that Andy finally worked up, as far as he got at Nixdorf, amounted to one program which read in data and did the first half of making this very simple little policy. Then you ran in the second part of the program to complete the job [laughs]. Evidently some insurance people got interested in it and used it. Andy and Thorpe talked together about it, and thought that this thing had real possibilities if they could only carry it further. They both became convinced that TRAC language was the basis for doing so. They decided to form a company and called upon me to join them. The company was called Data Concepts, Inc. and its offices were in Lexington, Mass.

As part of forming their company, they were able to interest Allstate Insurance in financing a prototype example.

Allstate played along and acted surprisingly creatively and constructively. Allstate financed the the procurement of a Texas Instruments 990 computer. This was a nice computer. It was an imitation of the PDP-11, that is, it had much the same structure. It had a disk memory. It had a tremendously large disk. The platter had the capacity all of two and one half megabytes, which contained all of the utilities, all of the compilers, all of the backup, and provided the room for data space for the application. The computer memory was 128 kilobytes. Sixty-four kilobytes was the size of the largest program you could put in.

The proposition was that I would transcribe TRAC for this project. In the early 1970s I had finally learned to program, and I had programmed TRAC for the PDP-10 using a model 33 teletype at my house in a remote access fashion using a modem. This was all done in assembly language, there were no high level languages. Peter Deutsch, of course, wrote his PDP-1 program in assembly. My PDP-10 program copied much of his internal structure and layout, and I carried it over to the PDP-10 as I programmed. Then, working with Thorpe Wright and Andy Diamond, I reprogrammed TRAC for the TI990. Eventually I reprogrammed TRAC again for the Motorola 68000 and for the Wicat computer. The Wicat organization (in Utah) is still in business. They are in the field of educational computer applications. They were in that before, and they took a whirl trying to become a figure in selling Motorola-based computers. This was before the big thing in workstations developed only a few years later. They were ahead of the times. They developed their own operating system which was sort of a ripoff Unix. They had an imitation "vi" editor, and so on. However, I programmed TRAC for the TI-990. Then when I programmed it again for the Motorola, I had some assistance from some students we got. They did a transcription from assembly language to assembly language, from one computer to another. My orders were "no innovations." Carry everything over exactly, functionally, completely, and accurately, just as it was done in the original.

Well, this business at Data Concepts finally had something like five or six major insurance companies as buyers to the method. The method was developed and it became very powerful indeed. This was just before the emergence of IBM's great mistake which was in bringing out the PC. The PC, of course, eventually destroyed the IBM company as it originally existed. They didn't realizethat at the time. However, when the PC appeared, I saw it very clearly. They didn't know what they were doing to themselves. This is one of the topics I listed as a vignette in computer history

to explore: namely, that IBM didn't realize what they were doing and why in 1981. Our Data Concepts insurance system turned out to be a system that worked, and worked very constructively and well. It was a kind of computer activity which IBM had tried to do, tried something like two or three separate times, both by itself and also in company with an insurance company, to try create a method of policy production. In all attempts they failed. We succeeded.

However, this little company, Data Concepts, that was set up and produced this product which worked, failed also. The reason it failed was one of the simple situations that you can get in business. The lead salesman, Andy Diamond, made the mistake in one of the first sales he made. He promised, I'm not sure whether it was to USF&G or some other company, that here's a prototype system, it calculates policies. It makes up policies completely printed out. In other words, by line printer output. It does the complete sort of thing that you need to do: make up forms, look up tables, and do all the calculations. Of course, he said, it doesn't exactly match the methodology that you developed with the quill pen over the last 100 years, but we have a very flexible system. You pay part of our price now, and we will modify our system to your satisfaction. Then you can pay us the rest in a balloon payment at the end, a half million dollars. Well, it turned out that at the end of this sequence they didn't pay. The same with several other companies. We went into Chapter 7 bankruptcy. That means everything vanishes. And why? Because they said, "Well, we won't pay you because we're not yet satisfied with it." "Well, what aren't you satisfied with it?" "Well, we don't know yet really. We can't tell you, but we aren't satisfied, so we are not committed to pay you."

TAPE 2/SIDE 2

MOOERS: Our working system used the TI-990 computer. Continental Insurance Company said, and all of them said, "Well, you're using TI computers. They're not IBM computers. They're no good. They aren't fast enough. Everybody knows that IBM computers are better even though they run slower. IBM is bringing out the PC. That'll solve all the problems of everybody because it's an IBM machine, and if you would just program TRAC for the IBM PC machine, all your problems will be solved, and moreover, if you do that, at the date that you've done that, we (Continental) will pay you $500,000 for the system with that new code." We (Data Concepts) were having money

problems. Well, it turned out that they held off the payment. On Friday the boys in the back room declared that they had completed the C language programming of TRAC on the IBM PC for the whole system. Now it turns out we had machine-independent type programming, and all of our base insurance scripts could be run indifferently on TI, Wicat, or PC. There was no change going from one machine to another. So the whole system was now running on the PC. So our software people declared on Friday that they had completed the transcription. The program was running. Continental was informed. On Tuesday the Chapter 7 bankruptcy was filed for Data Concepts. Continental had seen their opportunity and simply stonewalled and refused to pay. In other words, "if we (Continental) only hold out, we don't have to pay this amount. This company (Data Concepts) is going down the tube anyway" without the observation that if only they and USF&G had paid up what they owed and deserved to pay and the $500,000 that they had committed to pay, we would stay afloat. Well, they didn't, and the guru who was one of the investors, Herb Stein, who had moved in to Data Concepts, didn't handle the situation. The whole thing blew up, irrespective of the fact that the product was technically successful. So Thorpe Wright several years later purchased the dregs -- all rights and the software -- and he sold back to me all the stuff from TRAC with back licensing.

CHARLOTTE MOOERS: Though Thorpe wasn't a very good businessman, he was an honorable man, and he thought Calvin deserved it.

MOOERS: Data Concepts was in operation in one form of a company or another from about 1977 until 1984. It was dissolved in bankruptcy in March, 1984. At that point I was no longer owner of TRAC. I hadn't yet repurchased it. That didn't occur until about two years ago. I was no spring chicken at that time either. So what I did was spend a couple months in reformulating, rethinking carefully, the whole TRAC methodology and defining what an improved version would be, and did define such an improved version and then started programming it. So I reprogrammed TRAC into TRAC-2. So the first one was TRAC-64 which was TRAC as defined in 1964 and the TRAC-2 which was defined in 1984. The first parts of the TRAC-2 were running in July, 1984. It is still operative. It is, at the present time, very portable. It is running in Unix systems, DECs, VAX, and VS systems. And moreover you can transport any users' script from one machine to another: from a PC to a Unix machine to a VAX machine without changing. It's all in existence now.

In connection with TRAC-2, I fell in with a chap who got very interested in the intellectual achievement of TRAC and of TRAC-2 and he thought he could commercialize it. His name is Franco Vitaliano. Initially it was Vitagliano, but then he omitted the "g" later. Franco was able to catch the attention of many people with TRAC-2, but was not able to really sell it. I tried to cooperate with him and to be his development organization. Buying the machines that were required became very expensive. So I parted ways with Franco about two or three years ago. However, I still believe that TRAC-2, which carries the interest of TRAC-64, is still available as a very useful tool in computer science. I still have a complete copy of the C language transcription of TRAC-1, and Thorpe Wright has a complete copy of the insurance program which is called Simpl!e.

Recently, I've been spending time working on various aspects of TRAC-2, but after the work with Vitaliano, I haven't tried to open up an actual commercial enterprise, although I've been looking at certain opportunities that have turned up. What I was proposing was like many of my early ideas. For instance, in a network operation, there are certain opportunities for providing customer service in the way of descriptions of the Internet and so on. With TRAC, I demonstrated three years ago at BBN a complete system for taking a multipage article and displaying it page by page forward, and page by page backward, and it could be set up either on a PC, VAX, or Unix. We can either use a PC as a terminal and you go, say, over the Internet and run the TRAC program remotely and it sends pages to you, or you could get the documents downloaded to the PC and let the TRAC program and the PC run and display it. Not only open a document, but open one document, then open a second document, and flip back and forward and save your place and then search through the whole second document. Well, the lady in charge at BBN, her field was Slavic languages, and she wasn't terribly interested in just what the potential of any novel system like this might be. She was playing around with a Hypercard sort of thing set up on Macintosh. Of course you couldn't access the Macintosh over Telnet which you could with my system. Mine was something which was machine-independent and easy to update. You can't update features in a Hypercard system beyond the certain trivial level of the Macintosh Hypercard capability. I was in no position to push it hard. I didn't want to torpedo Charlotte's situation at Bolt, Beranek & Newman. There it was, all demonstrated and operating completely. It's still there and operating. The thing was put in place three years ago now this summer, and is still there in one of the sub-computers and running

and they have yet to find any use for it!

CHARLOTTE MOOERS: Well, actually if you have a Gopher system now you can do very similar things plus a few more things which could have been put into this, but weren't. This, again, was a situation where there was a window of opportunity which wasn't taken up, and then somebody else noticed that the opportunity was there. In this case the display of text over the network, which in turn depended on having rapid enough network access. The worldwide web, Gopher, and all of these, depend upon the fact that the network access is much more rapid than it was, and once that happens you can take advantage of it. Calvin's program was one that might have worked out, but it didn't.

MOOERS: To get something novel inserted into a dynamic situation like that, you've got to have somebody who has the time and energy to really ride it. It helps a great deal if you have an institution standing behind it like a university, and it helps a great deal if it's good. Well, if you have all of these things in abundance it doesn't have to be very good at all. It can be pretty punk and still go pretty far, like some of the stuff that's on the network now, or like Mortimer Taube's Uniterms, and so on. So this sort of thing unfortunately has been repeated a number of times in my career. I've been very much concerned with the, you might say, theoretical concepts of what I was doing and the depths of the reasoning and so on, and much less so in doing the infighting of selling of one sort or another, although I have done selling. But those are the ingredients and that's what happened. In other words some of my things have been successful conceptually and functionally and were not successful economically. Consider what happened to phlogiston theory of heat and other theories in the past. I think the interest in the history of science is in part of what were some of the alternatives that appeared. Why did it happen to them the way it happened, and what could have been?

In the way of TRAC, I think it has a continuing capability. That is, things like Windows and Windows NT are becoming so large and complex that I think that there is likely to become a backlash. An interest in using computer technological tools which are reasonable size which have a kind of versatility which allow you to do things that are not programmed into a given set piece of software, but which are meaningful. And people want to do this, either as a technician, or doing it for a one-time job, or to build on ad hoc add-on capabilities to some sort of standing utility. I

see TRAC as a very versatile and very capable vehicle that can do that.

Currently one of the projects I have in mind and am exploring is how to couple TRAC into the Windows capability which is called Visual Basic. This is a fantastic toy-colored interface. "Toy-colored" I say because intellectually it's very trivial. You get a lot of emotional impact from moving color, but intellectually [laughs] it's not that profound. The profundity is in the processes that you want to achieve. What is really going on? What is the mathematics? What are the elements being searched for? What are the elements being manipulated? Can they be presented? Can they be found? Can they be moved? And what mechanisms should do that? Well, Windows can't do these things, in part, because the people who built Windows initially didn't see any need. They only needed to try to provide a moving frame and a screen. You might say, Boolean algebra is not going to disappear overnight because we have a computer. And in the same way, TRAC has a kind of long-term versatility which is portable and is useable. And one of my projects at the moment on the one hand is finding a way to couple TRAC into some of these dramatic current visual developments without losing the inherent dynamics and logic of TRAC. In other words, without TRAC losing it's identity. And another project is looking for some young people who might be interested in picking up TRAC to run with. So I'm doing that back at home, and other things.

I began writing a history of TRAC and I've already put together quite a bit on that. So one of the targets, either in whole or in part, is *Annals of the History of Computing*. Of course the history of TRAC is more than just the bare history of the language, the story that you heard. There was a very considerable dynamic tapestry of interplay of ideas and technology and habits and so on which actually prevailed. That is, why didn't the insurance thing pan out? Well, it's not technology. The technology did pan out. So it's a complex history, but if you're really going to try to understand what the history is, what really happened, you should know that. So I'm trying to write that, in a history of TRAC.

There's also the, I wouldn't say "the history of information retrieval," because I'm only part of that history. What I can do is to put together, in a sense, my history, my involvement in information retrieval because in the library area there has been a long train of interest in this field, and my part is only a small part. But as a creator I am not the one

to put such a history together. I don't want to get involved in library research, in putting it together, and putting it in balance, and judging this fact against that fact. I think that what I actually did, and what I thought about, is meaningful to the history as a whole and I'm the only one who can tell it, and I should tell it. So that's the kind of history that I see myself putting together if I do something in information retrieval. I also have a large number of published documents which are not assembled in any conveniently accessible way, and I'm thinking of getting those reduced to digital form, and maybe getting them on a compact disc [CD-ROM] or something else. So these are some of the projects I had in mind right at the moment.

Recently for a birthday present, Charlotte let me buy a low cost, "obsolescent" 386 machine. We got it with 217 megabytes of hard disk, and initially it came with four megabytes of RAM and I just upgraded it to eight megabytes of RAM, so there I have a toy to play with. I recently got Microsoft's development package for C++. They call it Visual C++. In other words, it is a testbed for writing C++ programs. I also got their package on Visual Basic. The reason for the C++ package is part of the mechanism of tying TRAC into Visual Basic. To refresh your understanding, Visual Basic is a rather more thorough job of the original Apple Macintosh Hypercard. The original Hypercard was sort of a kindergarten kid's plaything. The Visual Basic is a deeper tool, and incidently, more complicated to handle. It can't be said to be a completed project because "OLE" -- object linking and embedding -- and a few other things which embody Bill Gates' vision of the future of multimedia aren't completely integrated in it. Yet a great deal is integrated. That is, I made a birthday card for Charlotte in color in Visual Basic in just a couple hours in the afternoon and I was just learning it. In other words, turn on the computer and the screen comes up with, "press this button," "press that button," and here is an Easter egg color greeting, "Happy Birthday Charlotte, I love you" sort of thing [laughs]. So that's what you can do in Visual Basic. My view is that you can only go so far in that. That is, if you weren't interested in sticking to show biz. If you want to go further, and do something serious and meaningful, you've got to couple into something else. I've seen TRAC as a possibility there to explore. So these are some of the current projects that I have ongoing which are added to the historical projects. I will send you the list of ideas for computer vignettes for your amusement and amazement. Just a little editing is needed for a probably 20,000 words or so of my story of the NOL period. That is my "Calvin at the NOL computer project." That is a lengthy version, not just the little piece than ran in *Annals of the History of Computing*. So that's ready to go out.

And the history of TRAC, I'm in the midst of. So you can see that in a sense, I'm up to my mischief as usual [laughs] and being kept out of mischief by these projects. But that's what's going on. This brings the story up to date with, of course, big gaps in details on the period of information retrieval. For which, one way or another, if you guys want to corner me and encourage me and learn something about information retrieval, that would be fine. So there it is. You can leave your recorder running if you wish because you have some set questions, I guess, or other questions.

CORBITT: The one thing we haven't covered would be the reactive typewriter and your work with that. Can you talk a little bit on the work you did on the development of that?

MOOERS: In the late 1950s it became clear to me that there were a lot of activities done by librarians which were done with typewriters and which were candidates for coupling into computer activity. The idea of the reactive typewriter was that the program in the computer would be essentially a partner to a person working at a keyboard performing an office task, and I used the library field initially as the epitome of the kind of the office task. As you know, with a library catalog card, each book is in embodied or represented by maybe six or eight cards and these cards oftentimes are printed and then they retype portions of the card at the top edge. They do it all manually and with great effort. So it appeared to me that one of the things that librarians could be helped with was a computer program to simplify this so they entered the material only once. Now, the Library of Congress project, MARC, could really be said to be an offshoot of my work.

TAPE 3/SIDE 1

MOOERS: There wasn't any machine of this sort that I was familiar with. Well, it turned out the Flexowriter did exist, but there was never a standard Flexowriter. Every unit had a slightly different code. I saw that with computers coming along it would be highly desirable to get cataloging information on paper tape in anticipation of computers coming into being, such tape would then be the basis of computerized catalogs. The idea of the reactive typewriter was sort of a general formulation of this idea of partnership between a keyboard operator and a computer. So in a sense it was a conceptual gambit. In other words, I described the idea, the function. I gave a name to it and started

an agitation. And, in retrospect, looking over the scene, it clearly was one of the elements that caught on because the Project MARC [of the Library of Congress], of course, is related to it.

I also started an agitation when, in the early 1960s, I gave a talk to the ASLIB, the British society of librarians, and also in Paris, France I talked to the French librarians about the "Tape-Typewriter Plan." In other words, the idea was to get hold of a tape-making typewriter and to start cataloging. The message was: "Here, look people. You are librarians. Do you realize that we have a computer revolution in front of us? You better start learning what a digit is, what the digital methods are, you better learn what the storage media are. It would be useful to practice and start cataloging, getting it into tape, because in the process you'll learn what's going on and you will educate yourself and it's all going to happen."

The reactive typewriter is a generic formulation of what is now known as a "terminal." The whole of the Internet is all reactive typewriters. And the Tape-Typewriter Plan was an early instigation to the library field to wake up and to realize that we are on the verge of a digital revolution involving your activities and computer capabilities and digits, and here is the way. You don't have to wait for a computer. You can start playing the game now and learn about it so you will be ready. So that was the message.

I also talked about a thing that was -- and you probably don't have mention of it because there was only one short paper about it -- the "Duffer Unit." That is, the minimal sort of device that can give you keyboard access to a computer, and it has to be user friendly. That means not so expensive nor complicated that a duffer can't use it. So that's why I called it a Duffer Unit [laughs]. So there's your answer. Not the short answer.

CORBITT: I've seen the term Zatopleg. Is that a variation on Zatocoding?

CHARLOTTE MOOERS: Yes, that was a term that we used.

MOOERS: Yes. Trouble is you are now talking to a mathematician or a logician or whatever, and I will try to give you

a careful answer that's true [laughs] and will stand analysis. Zatocoding gives a highly compressed representation into a physical media in a digital fashion, wherein each document is represented by a single card or a single record field, and the different index elements are then coded into that record field for the single document. To perform a selection you have to take the set of cards and serially scan the cards of the set, the coded field, one after the other to find a match or the fuzzy match. So Zatocoding involves a serial scan of the compressed digital coded record. A typical cardboard card that we used with these systems would have 40 digital positions in which all of the indexing information for that document would be impressed -- in only 40 bits. Now, typical collections might have 100,000 or 200,000 documents, and the vocabulary of indexing terms might run up to several hundred and with other extensions running higher than that. Well, with ordinary coding techniques you can't compress that much. You can't put that much coded information into 40 bits for a collection that large. Well, Zatocoding did it, but it involves a serial scan. Now, it is possible to perform a dimensional transformation of this situation in the following sense. Instead of a card with 40 positions on it and 200,000 card units in all, consider an inversion in this fashion. We consider 40 plates corresponding to the edge of one card. The coding field 40, is now represented by 40 plates, and the plates would be large. It doesn't take a very big plate for, you see, with 30 millimeters in each direction with one hole on each millimeter will give you a thousand locations in a square of 1 1/4 inches. So it doesn't take much of a plate to give 200,00 holes. One hundred fifty square inches would do it. A plate 10 by 15 inches, where you can locate holes. So where in Zatocoding the single card is notched on the edge, in Zatopleg we have 40 plates and a single document corresponds to one location. We code down that drill hole, as it were, for 40 positions for the coding field. Now we have one large physical object, the set of 40 plates, which corresponds to the whole 200,000 cards with coded edge. Instead of running 200,000 cards through a selector and looking at themas in Zatocoding, in Zatopleg for the selection we want to look at plates, e.g., at plate 9, 13, 15, 16. 17, 23 or so on. In Zatopleg, we just pull out the corresponding numbered plates, superimpose them and see where light shines through and find the coordinates, and that's Zatopleg. Each hole represents a document.

Incidently, nobody has done it as an operational matter. I've done it in experimental demonstration size. By this time in my history, when I conceived of the Zatopleg method, I was sufficiently cynical about the whole patent process, the commercial process, and the library situation, that I saw no mileage in trying to go anywhere with this. I took the

view -- this was early 1960s -- that, with the development of computers, these problems were going to get worse and worse and worse and we'll eventually need some clever methods such as I was working with. And among these things are not only Zatocoding and Zatopleg, but the little idea of "descriptors," which is another package of ideas, and is quite different. Now, the word descriptors has been adopted and has gone off in all directions. Before 1948 the word did not exist, of course, and was not in the dictionary. It's now in the dictionary and most people don't know that it was my neologism. I made it up because I wanted the new word to mean exactly what I described and, unfortunately, that never happened. That is, the word descriptor now means almost anything. The librarians never understood what I was talking about, which is an interesting psychological and professional phenomena and that's an another long story. So Zatocoding, Zatopleg, descriptors, reactive typewriters, lower case in ASCII -- all of these things I stirred up. So here you are, the good people of the Babbage Institute are housing 120 boxes of my stuff and I've never built a computer other than a Heath Kit H-89 [laughs].

CHARLOTTE MOOERS: The lower case in ASCII is something that's easy to explain to people [laughs].

MOOERS: However, they are surprised. "Wasn't it that way always [laughs]?" That's the answer I get back. But anyway, having achieved, that is, sowed these germs of insight and creativity into the field, I don't feel badly about asking you people at the Babbage Institute to store, try to make sense of, and weave it into the actual history of what went on. Imagine what computers would be if upper and lower case were separated? That is, if we had literary machines, and we had engineering machines, and never the twain should meet? Well, here in the 1990s we would probably be very forcibly trying to achieve a meeting of two kinds of computers, after a great deal of pain in between.

CHARLOTTE MOOERS: But much of what would have happened you can see when you look in the word processing field and see how exceedingly difficult it is to get the extended ASCIIs to work when you're trying to move things from one program to another. I mean people at BBN, which is supposed to be a leader in computers, do this type of reformatting all the time. It's terrible [laughs].

MOOERS: I believe it is true that IBM shot themselves in the foot by bringing out the PC. The computer as a "high

altar" lost its edge, a result it had taken IBM 15 years to build up. That wouldn't have happened if the PC hadn't been used as the vehicle for doing word processing, which was then using special machines. Even the PC can do word processing cheaper, and the word processing, of course, moved the computer into the front lines of the office. That whole transition wouldn't have occurred if IBM had kept the mainframe behind the plate glass windows with people walking by in hushed audience [laughs], walking by looking reverently over at the mainframe. The PC brought these things to the people, and lost IBM the mystique, and the lower case was the key element of coupling that whole thing into the activities of academics and word processing. Word processing was of value in the office and it was also matched with what was going on in academic circles, so the word processing brought all of these things together with main computer activity. But had they been kept separate, IBM would be a prosperous company still at the present time [laughs]. If this analysis is true, I was very material in changing the whole future of IBM. IBM and the other large companies were completely dedicated to having monocase, the upper case, for the computer code system. This would have stalled or delayed this whole process. Of course, you can't predict everything that would have happened. That is, to predict in 1967 for ASCII or in 1981 for the PC, that IBM would be in its present difficulties maybe five, ten years later as a result. These were the kind of elements that really worked their way deep into the fabric of the computer science scene that we see at the present time. I'm the "troublemaker" that [laughs] was very important in all of this. Not only in this, but also in the library science field. I stirred that field up, too.

[END OF INTERVIEW]