Oral History of Richard Greenblatt

Interviewed by:
Gardner Hendrie

Recorded: January 12, 2005
Boston, Massachusetts

Total Running Time: 2:33:00

CHM Reference number: X3056.2005

Gardner Hendrie: We have today with us Richard Greenblatt who has graciously agreed to do an oral history for the Computer History Museum. Tell us a little bit about your family background, your parents, where you were born and brought up.

Richard Greenblatt: I was born in Portland, Oregon, and my family then moved to Philadelphia, Pennsylvania for a couple years, but I actually grew up almost entirely in Columbia, Missouri, which is a university town halfway between St. Louis and Kansas City. At the time it was about 30,000 people or so and the University of Missouri had 15,000 or so university students, two other universities in town so it's much smaller than the University of Missouri but anyway it was a rather ideal place as I look back on it now <laughs> actually to grow up and it was a college town. There were— a lot of the teachers were wives of students or whatever and it was quite progressive in a lot of ways and I also felt that I came through at the right time. Just after I was through, they started in with new math and a bunch of other teaching things that I <laughs> didn't agree with at all. So I had the majority of my high school before they switched to proofs, quote, unquote, and other nonsense that I thought was useless.

Hendrie: What did your parents do?

Greenblatt: Well, they were divorced. My father was a dentist and had a practice in Philadelphia and my mother, when they divorced she moved back with her family. She had grown up in Columbia, Missouri, her family lived there, so we moved back with my grandparents and we had another house and I grew up there. I spent a lot of time at the University of Missouri. There was a student union there and so when I was very young, in the third grade and stuff, I would go to the university student union at University of Missouri and play chess with the university kids and so forth and basically I was pretty good. I would hold my own with them. It was before the days of ratings and sort of chess organization really penetrating out as far as Columbia, Missouri so I didn't have an official rating and so forth. Occasionally, I visited my father and whatever and we would visit the Franklin Chess Club in Philadelphia or something like that so—

Hendrie: Do you remember when you learned to play chess and how that happened?

Greenblatt: Not exactly. My father, I'm sure, taught me. He was fairly good. He had been a chess champion of— well, I think it was actually later. I think he had finished second in the city of Portland perhaps or something I think and later I think he won the city title in Eugene or something. He was kind of at that level. He wasn't a really serious player but he was pretty good—

Hendrie: And he enjoyed that. That was a vocation of his—

Greenblatt: Yeah, and he occasionally visited in Columbia and so he initially took me to the University of Missouri Student Union and they would play chess and bridge there so I learned both of those games and

I played quite a bit and so forth. So it was a rather unusual bringing up in that way. I was quite good in high school and I won a bunch of awards and did this and that and I probably— there's no use going into too much of that. Anyway, I got admitted to MIT and I arrived at MIT in the fall of '62.

Hendrie: Did you have any brothers or sisters and what did—

Greenblatt: I have one sister and she had epilepsy and so she lives in a small town in Missouri.

Hendrie: It was just the two of you.

Greenblatt: Just the two of us and really just me in some sense although she was there too but not—

Hendrie: You were obviously playing chess at a very early age. What did you think at that time you might be when you grew up, that you wanted to do when you—

Greenblatt: Well, I was into a number of things. Through the university I met some engineering students; I was involved in ham radio. There was a radio shop in the area and a man ran the radio shop and I hung out at this radio shop and read— he had a lot of literature of various kinds and I read a lot of that. I worked after school as a Western Union messenger and I was around the Western Union office a certain amount but I was sort of science oriented, engineering oriented and mathematically oriented. There were a few computers, very few computers in Columbia, Missouri in those days. Our high-school mathematics club visited the State Farm Insurance Company which had a regional office, a fairly large regional office, in the town and they had a card thing with a card sorter and trick decks and you could watch the cards make pretty patterns and so forth and then at the University of Missouri in the business school they had a real computer which was a vacuum tube machine and we visited and again the high-school class looked it over. However, I didn't actually get involved in computers at all at high school or really know what a computer was. However, my uncle was a professor of surgery at the University of Missouri and had a very long career there and so forth and so I also was involved around the hospital and around the animal labs in the hospital and quite a few different experimental things going on in the hospital I had some exposure to.

Hendrie: When you graduated from high school, where did you think of going to college or—

Greenblatt: Well, I applied to two places, MIT and University of Missouri <laughs> where I didn't make a big deal out of it. I had quite high scores on my college boards and stuff and so I got accepted at MIT.

Hendrie: Why did you pick MIT? Where did that come from?

Greenblatt: Yeah. Well, that's a <laughs> good question. Matter of fact, how I even heard of MIT I don't really know. I guess I'd read some literature somewhere or something or had some awareness or something but it just seemed like a place to go and I applied and got accepted—

Hendrie: With the kinds of things you were interested in, you just thought—

Greenblatt: Yeah. It was rather chancy. I could have missed it. Looking back on it, I don't know exactly how it would have happened but anyway, but yeah. So in the fall of '62 I arrived in Cambridge at MIT and it was something they called Orientation Week which was where you had fraternities, and fraternities sent you an incredible amount of material trying to get you to join their fraternity and so forth, and so I arrived on campus and got picked up by a fraternity member or something and it basically took me one day to decide I didn't want to have anything to do with fraternities at which point I then had <laughs> a week of time on my hands, and so I moseyed around MIT and I discovered various things including the PDP-1, which had recently been given to MIT by the Digital Equipment Corporation [DEC] and it was located in building 26 right next to the TX-0 computer which had also been given to MIT. That was the first computer with a transistorized CPU. It had been built at Lincoln Labs initially as a memory test computer and I also discovered the radio station WTBS it was called in those days. They later sold their name to Turner. <laughs> There was W1MX which is the ham radio station, I had some interest in ham radio, and then there was the model railroad club, the Tech Model Railroad Club [TMRC] in Building 20 and so I became somewhat familiar I guess with all of those places and saw Spacewar! and saw the PDP-1s and so forth and so on. So then as school started, of course in those days school was still quite regimented. I also arrived at MIT at what I consider in retrospect to have been the ideal time. It was just after the extreme grind began to be diluted. In other words, the famous thing at MIT where they would say look to the left of you, look to the right of you, one of you won't be here at the end of the year or something like that and it was just an extreme grind. Well, I got there just when they had very first began to dilute that so they- it was no longer- they were no longer attempting to make it the extreme- most extreme drive they- grind they possibly could but yet they hadn't really diluted it too much <laughs> yet—

Hendrie: —do some other things a little bit.

Greenblatt: Well, not much. For example, in those days all the students, freshmen, took exactly the same courses. You had only two small electives and other than that all your courses were completely regimented and in large classes and then recitations, smaller breakouts and one thing or another, and your quizzes were organized. You had a quiz every Friday morning so your week was completely organized. You'd have a quiz in one subject one week, a quiz in the next subject the next week and so forth, rotating, and they were every Friday and so on. So it was still quite, quite regimented. Anyway, I did get involved in computers. In addition to the PDP-1, one of my roommates was taking a course in civil engineering. There was a 1620 with Load-and-Go Fortran and a CalComp plotter and I was involved in helping him do with that with Load-and-Go Fortran and then I took two electives. One of them was called 641. In those days there were only two computer courses in the entirety of MIT. It was two real computer courses plus this other thing in the civil-engineering department but 641 was a freshman elective. One other course called 6251 was kind of a real programming course. That was it.

Hendrie: 641 was about what—

Greenblatt: 641 was a freshman elective and it was computer programming and it was actually quite a good course. The computer involved was the [IBM] 7090, the campus-wide— You programmed it batch programming with punch cards so in the basement of Building 26 [it] was all full of key punches and 407s and various card-handling things. You submitted your debt and you got your problem back and your grades and so forth and so I did that. There was a good library there. There was a reading room, and then there was also the PDP-1 and there was a bunch of stuff around- connected with that and then there was the Tech Model Railroad Club which also had an interesting library and a lot of stuff and quite a few people who were involved in things in various ways including Alan Kotok and a guy named Robert Saunders, Bob. He's still sort of around. You see him at the hacker conventions and so forth. Another person who was a big person in my life in those days in a certain way is a guy named R.L. Nelson. He flunked out of MIT and left MIT at the end of my freshman year but basically he and I, mainly him with me looking over his shoulder, designed a bunch of telephone equipment. The Tech Model Railroad Club had elaborate relay equipment including telephone systems and something called The System which involved connecting the trains to the operating positions through telephone equipment like crossbars and so on. So basically, I designed it and helped design with these guys looking over their shoulders quite a few different pieces of equipment, something called the standard connector which is a Strowger telephone that was switching over. It goes up and goes across with a two-dimensional ___________ switch and then we designed a— The whole system was replaced. The system at that time was located under the layout, the layout was plaster, the plaster was falling down into the relays and making it unreliable and one thing and another. So it was perceived that there needed to be a new system and this R.L. Nelson guy sort of designed the essence of it and that was actually built. And it was built out away from- so it wasn't underneath the layout but the cab equipment was still controversial and wasn't built and I basically in that summer after my first year and so forth essentially generated a prototype cab equipment and finalized the design and then some other people kind of doubled over to it but Kotok in particular. There was a power structure at the club and that's another whole story. I'm a little bit of a countercultural type I guess you'd have to say but—

Hendrie: What's cab equipment when you say?

Greenblatt: Well, cab equipment is the set of relays that are associated with an operating position. In other words, you have the block equipment which is associated with a piece of track, you have a crossbar switch which is a complicated electromechanical, two-dimensional switch as used in the most advanced forms of mechanical telephone exchanges, and then you have the cab equipment which is associated with the operating position and basically has to detect when it's time to advance, when your train is moving forward. Well, it's the most elaborate. In this case, there were only two relays with- or actually, initially there were three relays associated with each block which was a big reduction from the old system that did nine and so forth. So this resulted in a lot of innovative design and optimizing and eliminate this con- <tape break>

Hendrie: You were doing relay logic design.

Greenblatt: Relay logic design and then so this cab equipment wound up it was more complicated. I think it had eight or nine relays in it per cab and there were five cab positions and so forth. So basically five different people could be running trains on this large layout. The trains could go wherever they went, the system would connect the train to the operating position and as the train went around it would remain connected to the one operating position. Moments later another train could pass over the same track under the control of someone else and it would be signified by a different connection on the crossbar switch and that would be maintained and so on so it was a rather elaborate system. Another friend of mine in those years, my best friend actually in my freshman year, was a guy named— No. Actually, no, it was the sophomore year. Okay. So anyway, the summer of my freshman year I went back to Missouri and at that point time on the PDP-1 was incredibly tight. There wasn't any timesharing system, it was just a single-user machine. There was a signup list that would go up on Friday mornings I think it was and basically within a couple hours that signup list would be signed up 24 hours a day for the following week. So basically, to get time on the machine was something that graduate students and a few others did. As far as legitimately signing up for it, I was too low on the totem pole to be able to really sign up for any legitimate time. However, I hung around the machine some and got to know quite a bit and occasionally would have a few minutes when someone didn't show up for their time or they left early or one thing or another. However, I didn't actually get on the machine to really actually do anything my freshman year really but I did learn quite a lot about it and I started writing a Fortran compiler and I just wrote it out on a piece of paper and never really ran it. The faculty adviser at that time was a guy named Jack Dennis, Professor Jack Dennis who is still around and there was a guy named Peter Deutsch who was the son of an MIT professor and he was in high school at the time and he knew probably more about the computer software than anyone else at the time, a frequent thing of course, of young people <laughs> involved in it and wind up knowing more than the elders and so he was around. I knew him some although not too much but anyway- but he had for example written a Lisp that ran on the PDP-1 and I learned a little bit of the essence of Lisp and that and so forth. So anyway that first year I went back to Columbia and I spent the summer in Columbia and I arrived back at MIT—

Hendrie: What did you do over the summer? Do you remember?

Greenblatt: Well, I think I mostly just did the things I did in Columbia. I played chess at the student union and stuff like that I guess. I don't recall anything too much. However, then when I arrived back the fall of the following year, the place was in a tizzy and basically a guy named Stuart Nelson had shown up and again more or less doing the same thing I did and he showed up for the Orientation Week and after one day stopped that and he was a phone hacker and he knew a lot about phone companies and operators and so forth and MIT had a bunch of things called tie lines in those days and basically a number of them went to places like Lincoln Lab and Haystack Observatory and one thing or another and a number of those places had ways that you could get outside, just simply dial 9 or 1 in some cases or do other things and so basically it didn't take him very long to discover the fact that you could make free phone calls and so forth and of course in those days phone calls were extremely expensive and the idea of making free phone calls was an incredible thing. So anyway, he had also attempted to hook up some kind of a dialing device to the PDP-1 and gotten the campus patrol and of course it was basically a major tizzy. So anyway, I became friendly with Stu Nelson, he became my best friend and we were friends the following year and so forth, and he had quite a bit of experience. His father had been on the National Television Standards Committee, the committee that actually defined the color standard that's used to this day. His father had then passed away or something, I don't know exactly the timing of it, but he grew up in Poughkeepsie, New York, and had quite a bit of experience with the telephone company. He also had some experience with IBM and with certain IBM systems and so forth.

So anyway, we came back and I knew more of what was going on. By this time, there began to appear a few more PDP-1s. The AI Project led by Professor Minsky had submitted a proposal to ARPA and the guy who actually submitted this proposal was a member of The Model Railroad Club, a guy named Peter Samson, also an old hacker of some note. Anyway, another PDP-1 showed up in Building 26 and then not too long after that moved to Project MAC which was in Tech Square and basically Nelson and I, having some experience on the PDP-1, kind of gravitated over there. Two other guys were the actual student employees but they weren't too terribly good and one thing and another so basically Stu and I were more with it I guess you would say on some level. Anyway, we sort of gradually started helping various people who were trying to use those machines and this and that and then we wound up getting involved there and becoming student employees. It was before the days of UROP but it was more or less like that and so Stu and I and another fellow named Jack Holloway did a number of things. Project MAC which was in Tech Square had another 7090 computer that they had gotten and their whole business was to make a timesharing system called CTSS so they had a timesharing system. There were two other timesharing systems that were about contemporaneous and the three of these can be viewed as the first timesharing systems. Of course, that's something that was going on in a lot of places and so forth but there was on this PDP-1 at RLE [MIT's Research Laboratory for Electronics], Dennis and Deutsch were developing a timesharing system for that machine. That was a machine that had 4K of 18-bit memory and it had a fully elaborate timesharing system and a bunch of things. It was a five-microsecond cycle—10 microseconds per instruction in those cases and just incredibly slow by today's standards but actually we did it every—

Hendrie: —tour de force.

Greenblatt: Yes, and it had a display and all kinds of good things could be done with that display and so forth. So anyway and then there was another fellow named Bill Mann who worked at BBN who was also around The Model Railroad Club. BBN had a contract with Mass General Hospital and had something called the hospital system and so this was actually sort of a little bit commercial, or it could pass for commercial in those days and they had a rather interesting system and I got to know quite a few people at BBN. Many of them had been at MIT shortly before and had left by the time I got there — a guy named Dan Bobrow another guy named Dan Murphy, for example. After my second year—

Hendrie: Had you done anything with chess while you were here particularly?

Greenblatt: No. No, I had not done anything with chess at all although I knew Kotok. Of course, Kotok had done things with chess. However—

Hendrie: When did he graduate?

Greenblatt: It took him a couple years to officially graduate but his normal time to have graduated would have been I think the year before I got there. However, his adviser, thesis adviser, essentially had been McCarthy. McCarthy had left MIT and gone to Stanford and so I guess the way you'd have to say it is that a lot of the momentum connected with that project had essentially gone with him. If there had been people around playing chess and doing chess and so forth, I'm sure I would have picked it up. However, it just wasn't happening. Kotok was out at DEC and nobody knew anything about chess really and so I didn't either. I didn't do anything with chess. That's right. In those days there were two major conferences, computer conferences, each year. There was the Fall Joint Computer Conference on the West Coast and the Spring Joint Computer Conference on the East Coast and basically as a student employee or something, I had basically got sent to these conferences amazingly enough <laughs> and so it was in '65. This was a bit later actually. This was after I dropped out of school and did this thing with Adams Associates that we talked about and came back to MIT.

Hendrie: Maybe what we ought to do is follow that path and then we can pick up on the conference.

Greenblatt: Oh, okay. Okay. Well, let's see. I was so interested—

Hendrie: After your sophomore year. Okay?

Greenblatt: Yeah. When I first came to MIT, I was on the dean's list. It turns out that I was very well prepared and amazingly enough—

Hendrie: Even from high school—

Greenblatt: Even from high school in Missouri and so forth and I had to work fairly hard but on the other hand it really wasn't that difficult and I was on the dean's list. Okay. Well, however, I was getting interested in computers and I was getting less interested in school so my grades just went more or less in a straight line down. So basically, by the end of my second year I was on probation or something. I had-, and I think if we get the years right, that was when the AI Lab or the predecessor of it, it was part of Project MAC, but anyway they traded in their PDP-1 to DEC for a PDP-6, the new PDP-6, and the state of the software on the PDP-6 of course was quite primitive. So initially, DEC allowed us to keep the PDP-1 even after the PDP-6 was delivered and so we used the PDP-1 to edit on paper tape and so forth like that. Well, there came a day where they said okay, on [a] Friday- they said on Monday DEC is coming and they're going to take their PDP-1 and we go ohhh because well, if they'd really done that—

Hendrie: —had not all moved to the PDP-6—

Greenblatt: Well, not the system software in particular if we would have had to prepare programs on a typewriter- on a teletype machine which was even harder than a [Friden] Flexowriter. On the PDP-1 they had something called Flexowriters which was this kind of a mechanical—

Hendrie: A predecessor of the teletype.

Greenblatt: —and so on. Anyway, so basically, given that, Nelson Holloway and I went into crash mode for this weekend and we essentially re-implemented- and of course in those days a lot of the software existing on the PDP-1, the general idea was to implement improved versions for the PDP-6. So there was a program called TECO [Text Editor and COrrector] that had written by Dan Murphy on the PDP-1. He was associated with the Spark chamber group which was another group that had a PDP-1 and we had become users of that. We had made some patches to it and made it so it could use micro tapes somewhat and different things.

Hendrie: TECO is an operating—

Greenblatt: An editor. Well, it was— I can go into some stories about that if we want. I don't know how much we want to push down on this. Anyway, we decided we had to write TECO for the PDP-6 over this weekend. So, to make a long story short, just as they came on Monday morning to take the PDP-1 <laughs> away, we had PDP-6 TECO on the PDP-6 able to edit itself and it would go on from there and even at that time it had a display. Our PDP-6 configuration had what's called a type 340 display which is a fairly elaborate display with—

Hendrie: With a light pen and a bunch—

Greenblatt: Oh, it had light pen and it also had an increment mode and a character generator and some other things that were—

Hendrie: When you moved the software, did you write it in PDP-6 assembly language?

Greenblatt: Yes. We wrote it in PDP-6 assembly language. We wrote it from scratch. We didn't consult the— Actually, there was a separate story with each piece of software but with this TECO we just started out and just wrote it from scratch. Earlier we had worked on the PDP-6 Lisp. Okay. Yeah. Maybe I should go into that because I don't think we had this tape running when we discussed that. Well, somewhat earlier at The Model Railroad Club there was a time when they were working on the prototype PDP-6 at DEC. There was a lot of discussion going on about the instruction set of the PDP-6 and how to do this and that and then the PDP-6 had quite a nice instruction set and quite a few different things and such. So anyway, we got into a mode where Kotok would bring a group of us out to the old mill in the evening and we would—

<crew talk>

Hendrie: …story about that.

Greenblatt: It was great. I just remembered that my car was called the Blue Bird.

Hendrie: Okay. Good.

Greenblatt: A 1954 Chevy convertible. And what I did in those days, it was before there were Chinese restaurants in Cambridge much. So essentially we all got in the car and we came to eat in Chinatown, right here. And, of course, the central artery, we're right now looking at the remains of the central artery, which is almost entirely gone. But I spent many, many hours basically battling with traffic which, of course, was no where near as bad then as it is now, but it was still fairly bad. We went to Chinatown. We ate dinner in Chinatown. Then we went back and we hacked, and then at about two or three in the morning, we would very frequently come to Chinatown again and eat dinner.

Hendrie: Get some more food.

Greenblatt: Get some more food. And sometimes we would even do it for lunch. So basically I would make either two or three trips to Chinatown just about every day with a carload of people.

Hendrie: That's funny. That's great.

Greenblatt: And that was a hassle. And then at the very peak of it, when I was really in my most voracious consumption of food, I would actually eat four Chinese meals a day. By that time there was some Chinese restaurants in Cambridge, so maybe I would only go eat once or twice in Cambridge, and once or twice in Chinatown.

Hendrie: Wow. That's funny.

Greenblatt: Anyway, yeah, so we're.

Hendrie: Great computer companies, right?

Greenblatt: Right, right. So after the Joint Computer Conference of '65, it was basically the first time I had been to California, and California is a fantastic place, particularly in those days. I mean, the place was a garden spot, and you'd go into Palo Alto and you'd drive in on that Oregon Expressway and there's just plants everywhere and just completely unlike anything I had ever seen. We went out to what was called the D.C. Power Building. The Stanford AI lab was located in the foothills, sort of past the Stanford campus. It's on Page Mill and Arastradero Road, and it was an idyllic place. I mean, they had a really big building. It was built California style with outside walkways. Well, it was just an amazing place.

Hendrie: Okay. And this is where Minsky was now.

Greenblatt: No. McCarthy.

Hendrie: Oh, McCarthy. Yes, I'm sorry.

Greenblatt: Minsky stayed at Massachusetts.

Hendrie: At MIT, yes. I'm sorry.

Greenblatt: That's right. That's right. This is where McCarthy went. That's right. So that's right that I went out there and you're on the top of the hill, and you could see the lights of San Francisco and other things. Amazing. In McCarthy's office I pawed through various things. And at that time, McCarthy's Kotok program, which had kind of moved with McCarthy to Stanford, was playing the Russian chess program. Well Russia had an interesting story, too. There's a guy named Mikhail Botvinnik, and he was the world champion, and he was one of the strongest players ever in the history up to that time. Well, of course, there was a whole thing about chess in Russia. The Russians set this up as a test of their society versus ours. There had been a lead-up into World War II, where Russia had kind of had some good results just before World War II. Then there was World War II, of course, which was a tremendous connection for Russia. Then just after World War II, there was some famous encounters between the U.S. and Russia in chess, and won by the Russians. And the Russians had a group of grand masters that came along, led by this Botvinnik guy, who pretty well decimated the U.S. for many years until the era of Bobby Fisher. So basically Botvinnik had gotten involved in the chess program. And nothing too much came out of that, but they did have another program that somebody in Russia had written, and it played the McCarthy program, the Kotok program.

Hendrie: Now, it ran on a Russian computer?

Greenblatt: BESM-6, BESM-12, or something. That's right. It ran on a <inaudible>

Hendrie: Oh, one of the BESM.

Greenblatt: That's right.

Hendrie: Machines.

Greenblatt: That's right. And so, anyway, I was just perusing various offices, all of which were open and so forth, out there. And I saw some listings connected with this match. And so I set up my chess thing and kind of looked over and I was just horrified by the play.

Hendrie: By the play of both computers.

Greenblatt: Yeah. Well, both computers and particularly the quality of the analysis. Most of this printout was analysis from the Kotok program. And I also saw some kind of a textual thing, which I don't believe was Kotok's thesis, but which had some of the same information as Kotok's thesis. It was probably some kind of a technical report, or something, that was anticipatory to Kotok's thesis. Anyway, one of the things I remembered, and which I just talked with Kotok, as a matter of fact, a few days ago, was the detail that they had is Alpha Beta, and so forth, and they had these whips, and the whips were set at 4, 4, 3, 3, 2, 2, 1, 1. In other words, that was how many. It would first look at the top ply. It would look at the four best moves. The next plys, it would look at the three best. Next ply, two best, next ply, one best. Well, I just recognized immediately that that was incredibly wrong. You see, basically looking at only one wide, you just have no signals or noise function. In other words, you look at one move, which you think is the best, but there's a tremendous amount of noise. Well, you look at some more moves, and if you find that one of those are better, you've effectively rejected some noise. Well, essentially the thing that I knew that they did, they were very weak chess players, both McCarthy and Kotok. And basically they had a very romanticized view of chess. And so I knew, however, that chess is a very, very precise game. And really the name of the game is take the other guy's pieces, and you don't just go along. In any kind of a strong game, you don't just lose pieces, win pieces, lose pieces, win pieces. I mean, if you lose even a single pawn without compensation, then you may have drawing chances, if you're lucky. Otherwise, the game is lost. Losing more than one pawn almost invariably results in loss of the game, period.

Hendrie: Okay. And these would be things that you, as an experienced chess player—

Greenblatt: Yeah.

Hendrie: —at a reasonable level would understand.

Greenblatt: That's right.

Hendrie: And have experienced. And have scar tissue about.

Greenblatt: In some sense, that's right. Although I did not play in real tournaments, you know. I just played over the board chess with these university kids.

Hendrie: Yes, okay.

Greenblatt: And I had not played any chess at all since I'd been at MIT, so that was at that point probably three, three and a half years, or something. Anyway, I looked at this thing and I could see that the quality of the analysis was not good. And I said, gee, I can do better than that. And so I immediately set to it, and within just a few weeks, after I got back, I had the thing playing chess.

Hendrie: Now, what the thing be?

Greenblatt: The PDP-6.

Hendrie: The PDP-6.

Greenblatt: Yes.

Hendrie: Now this one you had written your own.

Greenblatt: Yes. I had written my own.

Hendrie: From scratch?

Greenblatt: From scratch.

Hendrie: Okay.

Greenblatt: And so then as word got around— Well, there was a guy a MIT in those days named Hubert Dreyfus, who was a prominent critic of artificial intelligence, and made some statements of the form, you know, computers will never be any good for chess, and so forth. And, of course, he was, again, very romanticized. He was not a strong chess player. However, he thought he was, or I guess he knew was wasn't world class, but he thought he was a lot better than he was. So anyway, I had this chess program and basically Gerry Sussman, who's a professor at MIT now, and who was also a member of our group, had played. It was around and it was available on the machine. People played it, and so forth. And basically Sussman brought over Dreyfus and said, well, how would you like to have a friendly game or something. Dreyfus said, oh, sure. And sure enough, Dreyfus sat down and got beat. So this immediately got quite a bit of publicity.

Hendrie: Yes. Word of mouth.

Greenblatt: Word of mouth and eventually later other forms of communication, as well. Anyway, so at that point, some other real chess players who also happened to be students at MIT, began showing up at the AI lab. And it so happened that there were three national masters who were students at MIT at that time.

Hendrie: Oh, my goodness.

Greenblatt: And so one of them, a guy named Larry Kaufman. He was probably the strongest one. He's a chess journalist of a sort to the present day. He lives in the D.C. area. He's an international master. He's not super strong, but he's pretty good. And then there was another guy named Alan Baisley, who which a long story connected with. And a third guy named Carl Wagner, who just played a couple of games. I really didn't get to know him that well, but he was also an MIT student. Anyway, all three of these guys came in and they started playing the machine. And I started getting in friendly with them. I started finding out something about what chess tournaments, and so forth, and what chess books were, and so forth. So, I had the idea. I don't know where it came from, but the way to gauge this thing was to play in a human chess tournament.

So it turned out that in those days there was something called the YMCU, not too far from here, actually. It's just off of what used to be the combat zone. It's on Boylston Street between the Public Garden, Boston Common and the combat zone. It's a little thing, a little bit like the YMCA, except that it was different, called the YMCU. Young Men's Christian Union. And it basically provided services to servicemen mostly, who were docked in Boston, or whatever they were. Anyway, they had a chess club, and they had a fairly large space, multiple floors, and a gymnasium, and quite a few different things in downtown Boston. So at some point, I found out about this and we decided, okay, let's play our computer in a chess tournament. So this was in the days before portable terminals of any kind. So for the first chess tournament, we got one of these big Checker cabs, and we put in a model 35 teletype, if you know that is. That's a big one, weighs—

Hendrie: Yes. Much bigger than the 33.

Greenblatt: Much bigger than the 33, weighs several hundred pounds, well, maybe a hundred pounds, and is basically a full load for the large trunk of a Checker taxicab. We took that downstairs. By this time there were quite a few people that were helping. This was a sort of a lab-wide activity, so I didn't do all this myself. But anyway, they hauled it up a couple flights of stairs and installed it in this chess club in downtown Boston. This YMCU was an old, stodgy thing, and had a funny phone system, so Nelson hacked their phone system. Again, these were old people that were very oh, I don't know, they'd be horrified by the fact that their phone system was hacked, but we managed to do something there, clip in a modem. One of my friends who had a shop, the shop was at the AI lab, and he was quite a machinist and had nails and so forth. He made a device that clamped onto a chess clock so that when you pushed the buttons on the chess clock, this also activated a switch and then somebody did some electronics so that this switch then controlled something called the restrain tone, which was something that was a part of very early teletypes. It involved slowing down the paper tape reader in case the machine couldn't keep up with it. So we had a detector for this.

Hendrie: For that tone back in MIT.

Greenblatt: Back in MIT set up. And so the computer could basically sense its time, and sense whether its clock was on or not. And we played. These chess tournaments are rather serious. In other words, it's held over a weekend. The basic time control, I believe, was 40 moves in- maybe 45 moves in two hours for each side. So the game would then take four hours. And there was five rounds scheduled, so there was three rounds on Saturday, and two on Sunday, or something like that. So that was sort of all day, night to evening, on Saturday, and then almost all day on Sunday. And so it was quite a big operation to do that in those days with the computer. We had people tending the machine, and typing in the moves, and getting things to eat, and dealing with problems, and putting paper in the printer. There was just a zillion little things that you had to worry about.

Hendrie: Was there any problem in just getting that amount of time on the machine, or people were really supportive?

Greenblatt: Yeah. There was no timesharing on the machine in those early days, so it was just.

Hendrie: So it was just hold the sign up for a complete two days.

Greenblatt: And this one machine was available for the whole lab, but on the other hand, everybody was sort of involved in looking at the game and whatever. So it wasn't really any work going to get done that weekend anyway, anyhow.

Hendrie: Exactly.

Greenblatt: So we first played in a tournament and it was just about a month and a half or so after I came back from that.

Hendrie: That trip to California.

Greenblatt: So that's right. So I think it was in December or maybe January.

Hendrie: Wow. Now how was this? You had a teletype at the thing.

Greenblatt: At the conference.

Hendrie: At the conference. Now was that actually connected to the computer?

Greenblatt: Yeah.

Hendrie: So if you put in— you would type in the move the other person made.

Greenblatt: That's right. The operator would.

Hendrie: An operator there.

Greenblatt: That's right.

Hendrie: And then the computer would figure out what the move was and print it out, type out what the move was. And then

Greenblatt: Yeah.

Hendrie: The operator would move the piece on the board.

Greenblatt: Yeah. And press the clock. That's right.

Hendrie: And press the clock. So there wasn't any manual transcription at the PDP-6.

Greenblatt: No.

Hendrie: Okay.

Greenblatt: No. It was automatic at that point. By that time we did have a line printer <tape break> of some of those, and so it was printing away, and you had to worry about maybe the paper in the line printer would jam or something, but not too often. On occasion. That's right. That was the very first one where we didn't have any telephone service. Now things went along. Things got a little more formalized. So for some of the following tournaments, we actually had our own telephone lines installed officially by the telephone company just for that <inaudible>.

Hendrie: So you didn't have to hack.

Greenblatt: So we didn't have to hack the thing. That's right.

Hendrie: The telephone. Tell me about do you remember what happened at that tournament?

Greenblatt: Yes. In the first tournament, the computer drew one game. It played five rounds and lost four, one of which was to Wagner, who is this master who, of course, didn't have any chance against him. But anyway, it did draw a game against a guy named Ben Landy. Ben Landy was a local legend who was an older guy who was involved in sort of setting up the chess tournaments and kind of being some kind of a thing for the chess club, and so on. And he was. His rating was, I think, about 1440, which is not too tremendously wonderful. Anyway, he drew. That was the first tournament. Then about a month later, we played another tournament and again another five games. This time the computer won one, and beat Landy, the same guy; beat Landy, and lost the other four. So then the listing I have here is for the third tournament.

Hendrie: These are all at the same place.

Greenblatt: Yeah. I think all of these were at the YMCU. Later there were a couple also at the Prudential Center.

Hendrie: All right.

Greenblatt: Somebody had a thing inside the Prudential Center.

Hendrie: But they're all local in Boston.

Greenblatt: Both were in Boston. That's right. And so this one I have here is the first one where the computer really had a positive score. I just looked over it last night. It looks like we got a by in the first round. For some reason, we didn't ply a game in the first round, so there's four games and it looks like the computer won three and drew one.

Hendrie: Very good.

Greenblatt: Some of these players weren't rated, so computing a performance rating wasn't too easy for that. But we then played in several more tournaments. The best single result, in my opinion, was a draw of an 1880 player. The way these ratings work, 1880 was what was called an "A" player. And he's really a pretty good guy. I mean, he's still not a couple levels below. There's sort of roughly standard deviations, sort of have international master. It's a different scale, but you can kind of view it as above 2400. Then 22 to 24, was national master. That was kind of in the range where Baisley and Kaufman and Wagner were. Then 2000 to 2200 was called "expert." And, again, pretty strong player. And then the next range below that, 2000 to 1800, was called "A." And the median player, I believe, was in the lower "A" range. And then it went on down by 200 points, and increasing levels of the alphabet. So I think by the time you got down to Landy at 1440, you were down to a "D" player, or something like that.

Hendrie: I see. Okay. Just out of curiosity, how are the numeric ratings established through?

Greenblatt: Well, there's an elaborate amount of stuff that's going, and the essence of the present system was in effect then, at that time. There was a guy named Elo, Professor [Arpad Emrick] Elo, who worked this out, and who has adjusted it over the years, and it's now become internationally standardized and so forth. So these are called Elo Ratings. And an early form of his system was in use at that time. There's been a little bit of rating inflation, I think, but aside from that, these ratings are more or less comparable with ratings from the USCF today. The computer, by the way, did become an honorary member of the Massachusetts State Chess Association and the United States Chess Federation, which was necessary in order to play into these tournaments.

Hendrie: These ratings, could you just explain just briefly the theory, how a rating is computed, or just the general idea behind it.

Greenblatt: Well, basically, there's approximately speaking, if you take the difference in rating between two players, then the probability of winning is some kind of an S curve. At a certain level, the lower rated player has no chance of winning at all. Then it basically kind of comes down as an S, and then at some point it goes the other way. So this interval is approximately 400 points. In other words, if a guy is 400 points above you, then statistically speaking, you have nearly no chance at all of winning. The way it works is that, in those days, I think it's a little more elaborate now, but basically given the two guys play, you sort of subtract their ratings and you kind of get what the probability- what the stakes are. So if you lose to a guy, there might be say ten points at risk. And if you are 100 points ahead of him, then if you win, you get only three points, say. And if you lose, you get minus seven, because you were expected to win and you lost, so it takes off more.

Hendrie: I see.

Greenblatt: But if you're way above him, if you're 400 above him, then you get practically nothing. You might get just one or two points, because you were expected to win and you won, and so who cares? So that's the kind of way it is. And in those days, it was whatever you win, the other guy loses, so you're actually sort of playing for a stake and to keep the system inflated, there's sort of activity points. Just for entering the tournament at all, you get a couple of points, or something like that.

Hendrie: Okay. And somebody worked out the math and it all appears to actually sort of be a stable system.

Greenblatt: That's right. And particularly now with more refinements than were in place in those days.

Hendrie: Okay. Good. Thank you.

Greenblatt: So, that's right. So the computer's performance rating for that tournament- given any tournament one of the things you can compute is a performance rating, which is what your rating would be. And I believe it was 1200 or so. In other words, a draw with a 1440 player, and then four losses. The best the computer ever did, which I believe was after this one, the performance rating, I believe, was 1820, or something like that. So the computer did get up into an A level. I was working on this, but it should be emphasized, of course, that this was kind of a sideline. I was also writing system software and managing and doing numerous other things.

Hendrie: You had a real job, too.

Greenblatt: A real job, and doing research and a number of things. And it was an awful lot of work to play in these tournaments. So I would guess we probably played altogether, probably in half a dozen major.

Hendrie: Now did you work on the program in between each tournament, in terms of trying to improve it?

Greenblatt: Oh, yeah. Well, particularly during those days, yes.

Hendrie: Yeah, during those days.

Greenblatt: Sure. People would ply the program, and then it would make bad moves in certain states in the game. And so then we would come in and try to figure out why, and put in things to improve it. And that's, of course, one of the central things about chess, as a matter fact. It isn't really appreciated by very few people to the present day, which is that it is so terribly precise, so it is so important to have the program debugged. So that is really the most important thing about the playing strength of a program. And so in particular, these computer versus computer matches, which they sometimes have, well, those are likely to be just random noise, at least until it gets to quite a high level, because the programs just simply aren't well enough debugged and if you get to a situation where, for one reason or another, one program does something bad, well that.

Hendrie: Yeah. Something stupid that wasn't in the- that's literally the result of a software bug as opposed to.

Greenblatt: And that's the reason why I never wanted to play in these computer versus computer matches. And never played in any of these world championship computer chess championship, or any of those matches, because it was just too random in the sense that the computer makes one bad move and then you're done for a whole year. You lose and you don't get a good result that tournament, you don't play again until the following year. So I thought it was much better to play against people, which number one, was sort of the goal from an artificial intelligence point of view. And number two, it's more available in the sense that anytime you want, you can go down to your local chess club and play in principal.

Hendrie: Could we roll back just a little bit, back to could you tell me about some of the things you did when you wrote your program after looking at Alan's.

Greenblatt: Well, see, I didn't look at his.

Hendrie: You didn't really.

Greenblatt: I looked at the printout of the chess analysis, but I did not actually look at his program.

Hendrie: Okay. Okay.

Greenblatt: And I didn't discuss chess with Alan, and so forth. Again, even though I knew him well, the whole chess thing just wasn't happening at MIT at that point.

Hendrie: Exactly. And he wasn't involved in it anymore, either, at that time.

Greenblatt: That's right.

Hendrie: So talk to me about what you did in your program that you think improved over.

Greenblatt: Well, I did have quite a bit of stuff. I mean, I had a plausible move generator. It had quite a lot of databases of square control and pins, and attacks and masking attacks, and ability to estimate the swap out that would occur on a particular square. How many pieces from the two sides are attacking that square, and so forth. This is quite a lot of databases, and so a lot of this was maintained incrementally so that as you make a move, you have to subtract one from all those squares, which you're no longer attacking; add one to all those squares which you are attacking, and do that for several other databases involved in that, too. So that was quite a bit of it. But it had a plausible move generator. It had a quiescence detector.

Hendrie: Now what was that?

Greenblatt: Well, that is basically where you consider all captures. I mean, the basic idea is that you consider the best so many moves as a function of depth in the tree down to a certain depth. And then at that point, what you're trying to basically do is come up with a representative value to return and to return into your minimax alpha beta.

Hendrie: Yeah. This is the evaluation problem.

Greenblatt: That's right.

Hendrie: How good is this position really?

Greenblatt: That's right. And however, the problem is, of course, that if you're in the middle of a capture, or in a situation where one piece is hopelessly trapped and can't get lost, can't be saved, then you're at great risk of returning a incorrect value.

Hendrie: Oh. Because you really needed to look just one more ply ahead, or maybe two, to understand what's really going on.

Greenblatt: That's right. And in particular, if it's a swap out, then it's just absolutely necessary.

Hendrie: By a swap out you mean your trade.

Greenblatt: Queen takes queen, pawn takes queen. I mean, if you look at queen takes queen, oh, I'm a queen up. Well, of course, you're not. I mean, pawn takes queen is coming right back. So therefore, if you're in the middle of a swap out, you absolutely must look at all the captures and recaptures and get to a quiescence state there. And even if it's not in the middle of a swap out if you have, for example, two different pieces being attacked, then there's considerable danger that you might lose one, because, you know, you only have one move. You move one, and then pawn takes the other one, perhaps, or something. So I had things like that that basically did look for possible advantageous swap outs, and if there's more than one of those, then it says, okay, well, we better look deeper, because we might be able to save both of these pieces.

Hendrie: So you would get down to however many plys you were going to, in terms of looking at all the possible moves, and then rather than just evaluating at that point, the quiescent goes.

Greenblatt: That's right.

Hendrie: And looks for these situations that are going to make the evaluation criteria bad, or lied to you, and goes a little bit further.

Greenblatt: That's right. And then there's also some technique involved in assuring that the position, the values are as comparable as possible. In other words, you are comparing positions where it's your turn to move versus positions where it's his turn to move. Well, you see, that's a problem because, on the one hand, if you just compare it as a static position, you won't get a comparable value because he gets a move, and he moves a piece to some advantageous square, and then now controls more squares, and so forth and so on. So that's good for him. So you've got to try to estimate what the amplitude of the oscillation. We're talking about a minimax. One player is trying to maximize. The other player is trying to minimize. So therefore, the general value of the function tends to go kind of like so. You need to try to estimate what the amplitude of that oscillation is and kind of average it together so that you get a value that's- from the odd plys, that is reasonably comparable with the value from the even plys.

Hendrie: Ah. Okay.

Greenblatt: And so on like that. And oh, quite a few other things, actually. Sometimes you have a long sequence of moves which however are forced. In other words, the fundamental thing you have is an exponential growth. So, you do have a depth's parameter. You can set it to higher values, but if you do, things will slow down generally at around a factor of six or so, per depth. In other words, there's around 36 or so average moves, and the minimax thing more or less cuts it down as a square root, so more or less it's comparable to an exponential tree growing a factor of six at each ply. So there's quite a few actual techniques of various sorts that are involved in that. <tape break>

Hendrie: Any others that just come to mind?

Greenblatt: Yeah. Well, this thing where if you have a forcing thing, then what you sometimes do is play out to the end of that variation, and then do another short search from there.

Hendrie: Okay.

Greenblatt: So this allows you to just sample a small part of the tree which, however, is quite a bit deeper than what you were going to be able to look at otherwise. And this is a good thing to do.

Hendrie: Yes. So the fact not just having a single layer, you just go down certain number, but in certain cases you go deeper.

Greenblatt: Now, just to talk about what our hardware was at this time, we were running on the PDP-6 computer. It was approximately ten microseconds per instruction. We did have the so-called Moby memory. The original configuration of this machine delivers from DEC, had 16k of 36-bit words. This memory was two relay racks big. It cost $120,000, two microseconds. However, in a famous interlude, at some point ARPA said, okay, you AI labs doing work, mainly us at Stanford, and maybe Carnegie, a couple of others, mostly us at Stanford. They said, okay, we've got an extra 400 grand, so you can buy a big, new toy. So basically we bought something called the Fabritech Mobile Memory. This was a 256k memory, 40 bits. It had two wires going through each core. The normal core memory had four wires going through each core. And it was two big bays, so it was a little bit larger, but it was a little bit larger physically, but 16 times larger logically. And at the time, was one of the very largest memories in the world. Who knows if there was some other one comparable some way. But it was vastly cheaper, so we had a special deal with Fabritech to buy it. It cost us just under $400,000. Stanford took their money and they bought something called the Librascope disc, which was a huge.

Hendrie: Huge. Yes. I think we have one of those at the [Computer History] Museum.

Greenblatt: And so on.

Hendrie: It's amazing.

Greenblatt: So that was a whole story.

Hendrie: But was this still a two microsecond cycle time memory?

Greenblatt: Yes.

Hendrie: This Fabritech?

Greenblatt: That was a little slower. It was a little slower.

Hendrie: I was going to say, you had to give up something when you went to a four-wire scheme.

Greenblatt: It was 2.8. Also, these memories were on a memory bus and they were asynchronous, so when configuring this you were sometimes concerned by the access time. It was a certain length of time when you got the data back. Then there was another time where the memory is cycling, writing the data back and so forth, before it's ready to accept another. So the Fabritech was a little bit worse in that. Not only was it 2.8 microseconds, but also the data came a little bit later in that 2.8 microseconds.

Hendrie: All right. But the PDP-6 had the ability to deal with an asynchronous.

Greenblatt: That's right.

Hendrie: Main memory.

Greenblatt: It had this memory bus.

Hendrie: Yep. Okay.

Greenblatt: Yep. So it was about 1/8th of a mip, is another way of looking at this. And what our search depths were, as I was just looking on this listing, for tournament play, our basic depth was 15, 15, 9, 9, 7. So at the first two plys, we would look at the top 15 moves, plus captures. 15, 9, 9, 7. And then 7, however deep you have to go due to these quiescent searches, you keep it at 7, but you never go below 7, so the thing with the Kotok program originally had with the 4, 3, 2, 1. I mean, of course, their searching was much slower, so they couldn't set it much wider than that, apparently, but just having it at any ply, a level of 1, you're just introducing almost pure noise.

Hendrie: Yeah. Because there's no decision.

Greenblatt: That's right.

Hendrie: What's it tell you?

Greenblatt: You're just taking what appears to be the best move and that's a very fallible estimation.

Hendrie: Yes.

Greenblatt: And then you're not looking at anything else. So that's what I was just telling Kotok the other day, and he pulled out his thesis and sure enough there that was. And I told him I thought that his program would have played much better had he simply set it to 7, 7, and set the depth to 2, instead of 4, or whatever. And he said, well, they basically were not sufficiently good chess players to really understand that that was required.

<crew talk — end of tape>

Hendrie: Okay, one thing I had wanted to ask was the— Ken Thompson, you know—

Greenblatt: Yeah.

Hendrie: Quite a long time later [he] became pretty interested in—

Greenblatt: Yeah.

Hendrie: —in chess and he—we made a lot of use of—he actually made these—uses of databases of historical classic opening moves.

Greenblatt: Oh yeah, no we did too.

Hendrie: Okay. Could you maybe talk a little bit about what you did there?

Greenblatt: Yeah, we had that right from the beginning, or nearly the beginning. This guy Larry Kaufman generated our opening book. And I had a thing that encoded the opening moves and in a small amount of memory so that it fit in the computer memory and so forth. And he—so he did that.

Hendrie: And you put in main memory?

Greenblatt: Yeah.

Hendrie: Okay. So it was loaded in main memory?

Greenblatt: That's right.

Hendrie: Alright.

Greenblatt: And it was compacted pretty efficiently. By later standards it wasn't so big, but at the time it was pretty good sized. I don't know I think it was probably 8,000 or 10,000 moves in there.

Hendrie: Wow.

Greenblatt: Kaufman spent quite a while doing it. Of course, a number of the moves, you know, would—the game would just play it. There was very little chance of it actually staying in the book. But I think

Kaufman did a good job and he did—we did perceive some of the — well of course, one of the basic things about computer programs, really to the present day, is that they're very tactical. It's much easier for them to see that the tactics than the strategy. So what's called a closed position is hard for a computer. That's where all the pieces are blocking each other and it's very long maneuvering. Whereas, an open position which is sort of tactical combinations is much more to the computer's liking. So therefore, when designing the opening book, you want to kind of play offbeat and kind of unbalanced type openings that tend to lead to these closed position—open positions, which are then good for the computer. And so we realized that and I think Kaufman did a fairly good job of—

Hendrie: So there was—you did some strategic thinking in figuring out what to—

Greenblatt: That's right. But it—

Hendrie: —what is the opening positions are?

Greenblatt: That's right. It's not based just merely on the pure logic of chess of what is objectively the best move. What you're trying to do, of course is to get into positions where the computer will play its—will understand "the position" and play its type of a game to the best advantage.

Hendrie: Alright. About how deep do the opening, you know—and if I give an opening—

Greenblatt: Well it varies tremendously. I mean in many cases—

Hendrie: Would you—

Greenblatt: —of course there's a certain game that is played and frequently it's just easier to just type in the whole game or pretty deep into the game, rather than cut off someplace. But that is just, you know, that is just sort of a gratuitous at that point. It—but in most positions, as a practical matter, it would stay in the opening for opening book for maybe six, seven moves.

Hendrie: Oh alright.

Greenblatt: Not too many.

Hendrie: Yeah, but still.

Greenblatt: Sometimes it's—

Hendrie: <inaudible> number.

Greenblatt: Yeah. Sometimes, and of course it also—

Hendrie: It depends on the opponent? The opponent goes out of the—

Greenblatt: Absolutely.

Hendrie: But then—

Greenblatt: And the opponent plays pawn the queen, rook three then you're immediately out of the book.

Hendrie: Yeah.

Greenblatt: Right, first move.

Hendrie: Yeah. Okay.

Greenblatt: So.

Hendrie: Okay.

Greenblatt: It also has something to do with white versus black. Of course, the white is trying to get a position and get an advantage. The only thing that sort of makes opening books for humans finite at all is the fact that white is trying to get an advantage and black is trying to equalize. Once black is equalized then that's the end of it. So as far as the opening theory is "concerned" for humans.

Hendrie: Okay.

Greenblatt: So, if the computer is playing white, well again it can just play a completely random move such as pawn the queen three, it will immediately be out of the book—actually pawn the queen rook three let's say.

Hendrie: Yeah.

Greenblatt: And it will still be an even position. So with white, it's very easy to get out of the book in an even position. With black, on the other hand, you know black is kind of the underdog trying to creep up. So if black just plays a random move then there's more of a danger of white getting a significant advantage out of this.

Hendrie: Yes.

Greenblatt: And actually, having—maybe a tangible advantage. So—

Hendrie: Okay. So what about end game, when it comes to relatively few people have done a fair amount of work?

Greenblatt: Yeah. I didn't—

Hendrie: In analyzing those even using computer analysis.

Greenblatt: Well I just did the bare minimum that was necessary to know about pass pawns and if somebody has a pawn that's running on, you know, you don’t want to stop without giving that due course and so on and so forth. But I never did anything serious with end games. Some work had been done. A gal named Barbara Huberman who later married, I forgot what her name is—anyways she was around for many years. I never really met her. I heard about her, but she had done some kind of a classical end game thing. But these come up so rarely in practice that from a practical point of view you have to be at a very, very high level of play before there's really any significant chance that that will improve the strength of the program.

Hendrie: Okay. Alright. Good. What else should I ask you about, you know, you're—

Greenblatt: Well—

Hendrie: —design of the program and the structure? What other things?

Greenblatt: Well let's see. What we could say is that after that first year then I returned in '66 at the Fall Joint Computer Conference and I generated a paper. So in the fall—in the proceedings of the Fall Joint Computer Conference of 1966 there is the paper that I wrote about this thing, and I wrote it with—

Hendrie: Oh really, okay.

Greenblatt: —a guy named Steve Crocker who was a graduate student at the time. Later went on to become a <inaudible> and various things. And after the mid-60's, say '67 or '68 I — the chess program was there and I would kind of toy with it occasionally. But it wasn't really that large a fraction of my overall effort, because I was doing so many other things, involving system software and later Lisp machines and so on.

Hendrie: Okay.

Greenblatt: However, we did play, occasionally in a chess tournament or occasionally somebody would visit or occasionally there would be a match set up. We did play Ken Thompson once, he won.

Hendrie: With his bell?

Greenblatt: With bell.

Hendrie: Okay.

Greenblatt: We played a couple of games by ham radio with a program in Switzerland, we won that. There—when the Papert bet up after ten years, you know, essentially Papert and this guy David Levy had made a bet that a computer program wouldn't beat him in ten years. And—

Hendrie: David — it wouldn't beat David Levy?

Greenblatt: Yeah. And so David Levy won that bet, however he was only ahead by about three or four years, you know. David Levy was an international master, so he was you know a strong player but not on the world stage, he was not that strong.

Hendrie: Mm-hmm.

Greenblatt: And there's— I would guess there was probably something like at least a 100, maybe a couple 100 grand masters who, you know, the next —

Hendrie: The whole next —

Greenblatt: — that's right.

Hendrie: Two hundred points.

Greenblatt: So basically I think that bet came due in approximately 1980 or so and probably six or seven years after that was when chess programs kind of passed him by. I don't know if he ever actually played a match with a computer program that he lost. But he would be a definite underdog of course these days.

Hendrie: Yes. Okay.

Greenblatt: But anyway, we played that and we did a few things. Then I guess we can skip forward a few years.

Hendrie: Okay.

Greenblatt: And in the early 70's we had gotten to the point where the PDP-6 and then the PDP-10 address space, which was 256K, was no longer enough. And, you know, but programs were trying to expand and there wasn't enough room because you're running out of address space. And so we realized also that there's a lot of things can be done. So it became clear that we were going to have to do something else. I started the Lisp machine project in 1973. And for about a year I worked just by myself on an interpreter that ran on the PDP-10; it was a software emulator; emulated a macro code. And the macro code was designed especially for Lisp, the tag machine and so forth. During approximately the same time frame, [J.C.R.] Licklider, you know— there had been a split of the AI Lab and the so called laboratory for computer science into two laboratories. And Professor Licklider, who had sort of been head of the LCS decided he didn't want to do it anymore. He was getting old and he wanted to do his research and so forth. So it came up that a guy named Ed Fredkin, who had been around the AI Lab —

Hendrie: Ah, yes.

Greenblatt: — and Fred ____________ and so forth was kind of standing there and he had been — had had a successful company and I had known him, and my friend — a student that had also worked for him and so forth at Information International [Inc. commonly referred to as Triple-I or III]. Anyway, he was sort of tapped to be head of LCS [Laboratory for Computer Science].

Hendrie: Okay.

Greenblatt: And he was a capable guy. However, he was — he wasn't — it wasn't a very good fit actually. And he has his island in the Caribbean that he likes to go to and he disappears so forth and so on.

Hendrie: Yes.

Greenblatt: So, but at some point he also had an interesting line of research having to do with so called digital physics. The idea that physics is explained by some kind of an automaton machine and some sort of time. So a particular guy named John Moussouris was a Rhodes scholar at Oxford. And he was doing his PhD and he kind of bogged [down] and he decided that this digital physics was interesting and he would come to MIT —

<tape change>

Hendrie: Okay, so. Let's see, where did we sort of stop? I guess probably after your first sort of work on chess. Maybe you could just sort of highlight what things you worked on up to your involvement in Lisp and Lisp machines and that sort of thing.

Greenblatt: Well it was all going on in parallel.

Hendrie: Okay.

Greenblatt: — I worked on all these things. And Lisp actually — well this whole group at the Mala River Club of course had done this original Lisp on the PDP-6 and the prototype at DEC. And then after we got a machine, you know, we kind of picked it up and Stu Nelson and I did a Lisp compiler and that was a big deal in those days.

Hendrie: Okay. The first system had been just an interpretive system?

Greenblatt: Yeah, yeah. There had been a compiler before on the 7090 system but — and some other ones, not local, but ours was quite a bit fancier.

Hendrie: Okay.

Greenblatt: In various ways. So anyway, yeah, that was a major project and that of course was our primary resource — just about all the programming in the lab and in maxima and so forth was done in Lisp by using that compiler. And then I sort of passed on and various people took it over. I just poked at it a little bit, but too much for a while. But I guess really the thing I did mostly for quite a while was the time sharing system and you know it was just a lot of things; different I/O devices and different developments, because a lot of things had to do with time sharing system.

Hendrie: Now how did that time sharing system get started sort of in parallel with the Project MAC?

Greenblatt: Well the one that was really in parallel with was the DEC one. I mean you know we got our machine and DEC, you know was developing a system but it was very primitive and had a lot of things we didn't like about it. So—

Hendrie: If it had been really cool you might have just said, oh why do we need to do this?

Greenblatt: Yeah.

Hendrie: But it wasn't?

Greenblatt: But it wasn't. Actually, initially, we didn't use a time sharing system. Initially, it was just single user and of course at that time it was only 16K of memory on the machine. What could you do? But then we did get the Moby memory, the 256K, and of course that was when it became possible to do serious time sharing and stuff.

Hendrie: Now were there drums on this machine? Was there backings?

Greenblatt: Well that's another —

Hendrie: Random access backup store — backing store?

Greenblatt: That is a long story. And we can go into it. It's fun. But it will take us a while. Well, the short answer, is initially no. And it was a real problem over the years. The way it developed was that there was a company called Analex here in Boston. And they had made printers and at some point they decided to try their hand at disk drives. And it was the days of the IBM 2311, but these were very expensive. They were only available on rentals. The interface wasn't available; there's no controllers. So you know, we were hesitant to do that so — but DEC didn't have anything that was reasonable that you could buy from DEC at all. So we decided to get this thing from Analex — the thing from Analex was really quite a kludge to this day. It was based on a mechanical D to A. I don't know if you seen these things — sort of a thing on the wall. It consisted of a shaft that went down this way, with clutches and gears and little wheels that had two D10's. So the thing had a clutch and it could either go to stop in one of two stable positions. And then sort of a like a locomotive, like a steam locomotive thing, there was shafts and balance beams, so forth connected to these in such a way that it affected a mechanical digital to analog converter. And there were ten of these wheels, so each of one could be in either of two positions. So basically it had the unusual characteristic, which they thought was a great thing, was that the seek time was independent. You know most disk drives you seek from one track to the next and then you get to the near one —

Hendrie: A lot faster than the far one?

Greenblatt: That's right. Well this idea was interesting— but that was not the case. That it took a certain time for these wheels to make a half revolution. But once the wheels had made a half revolution then the balance beams did their work and the heads were then in the right position. So the seek time from any track to any other track was a constant.

Hendrie: Okay.

Greenblatt: Well so we got the raw disk drive from Analex and we built our own controller for it; at Holloway — meaning they built it. And we didn't know anything. The machine, it turned out, was just outside the door of this machine shop. This guy had dropped out of MIT and went to work at Lockheed as a machinist. And you know, he would just — had this milling machine and he would just really feed the metal in there and put in the heavy oil, you know, and the smoke would come out of it. This poor disk drive was just sitting just outside the door of that machine shop.

Hendrie: Oh.

Greenblatt: We didn't have the faintest idea of what was involved as far as —

Hendrie: Contaminants or anything?

Greenblatt: — contaminations or — and neither did the Analex people, really. I mean they had a little — some air cleaner but it just was totally inadequate. Needless to say we had disk crashes. Just then, Analex had been kind of getting pretty old in the tooth as far as their printer business and they hadn't been keeping that up too much. And they — another company in particular called Data Products had come out with a better printer.

Hendrie: Yes.

Greenblatt: So basically we didn't accept it and Analex decided that they were going to get out of the disk business and so we sent it back and we didn't have disks for— well we didn't have any disk at all for a year after that. We just had micro tape, the device for— you know it was 65K of 36-bit words on a little tape, yea big <gestures> and it took a couple of minutes to wind.

Hendrie: Right.

Greenblatt: So —

Hendrie: Could not make a paging story even if you had the idea.

Greenblatt: Yeah.

Hendrie: Out of that.

Greenblatt: We actually did just sort of think about that just a tiny bit. But of course it was totally impractical.

Hendrie: Yeah.

Greenblatt: So then there was another company called Data Disk. And they had a kind of a player type disk that's kind of a large Victrola desktop unit and it recorded on some hard surface. Or it was actually a contact recording thing. Basically, we made another controller and software to interface that thing and we used it for a number of years. But that thing had the problem that it was very small. They — two of them together — we had two units — the two of them together could dump the Moby memory once. So each — the entire capacity of the unit was 128K, 36-bit words.

Hendrie: Yes, alright.

Greenblatt: So again, it was — and it was quite slow. But at least you could have the temporary stuff with the data that you were working on while you were on the machine. At the beginning of your session and the end of your session, you read — wrote for micro tapes. But in the middle of your session you could use that and then we did this 256K main memory at that time. So that was pretty big and allowed you to. And then finally we decided well we really are going to have to go this 2311. By then IBM had kind of loosened up their policies on 2311's a little bit. And so Stu Nelson, who by this time, was in the Systems Concepts Company, but was still good friends, he and me — sort of mostly him, made a 2311 controller. And we actually got some 2311's and used those. And then finally, Systems Concepts decided to develop a sort of a more of a heavy weight IBM controller. It turns out that this was an endemic problem. I mean DEC had no peripherals. Anyone who wanted to use DEC machines for serious work was just at a great loss for peripherals because the tape drives and the disks and so forth were just not commercial quality compared to the IBM ones of the day. So anyway they made this disk control and we used that. And then finally, they made another product called and SA10 which was an IBM channel interface. And actually interfaced not directly to the disk drive itself but to an actual IBM channel which then meant it could interface to most any IBM device. And that turned out to be in incredible product for them. They sold those things for ten years or something. Then they sold a great number to a lot of DEC customers and so forth.

Hendrie: And that allowed you then to — if you had the money you could buy —

Greenblatt: Buy the —

Hendrie: — commercial grade —

Greenblatt: Yeah. We however didn't do it.

Hendrie: Of course.

Greenblatt: We had this one that was interfaced just directly to the — to what by that time was called 2314's. And then for a little while there, there was actually a competitive start up company where a bunch of different companies started making their own 2314's and going into commercial —

Hendrie: Yeah.

Greenblatt: — competition and so forth. So we — there was a company called Calcom that made — and so for — we got from them and basically we wound up getting double density 2314's —

Hendrie: Okay.

Greenblatt: — upstairs and downstairs and we had a full eight of those. And that was a fairly decent —

Hendrie: A robust amount of secondary storage —

Greenblatt: — of secondary storage.

Hendrie: — for a time sharing system.

Greenblatt: Yeah. And that's right, so.

Hendrie: Good.

Greenblatt: But —

Hendrie: Well I interrupted you.

Greenblatt: That was over a number of years. I mean each of those cycles of designing are hard — a piece of hardware and building and writing software for it and so on, took at least six months and more like a year or sometimes almost a year and a half or two years. So you know, just years went by and progress was gradually made. Also on the display front, there's likewise a story. We had teletypes and Tom Knight made a teletype multiplexer and then there was a totally funny GE system that we got and we interfaced and then several things. And then finally one time McCarthy came to visit on sabbatical or something and we got into —

<tape change>

Greenblatt: … that's of interest is the software. There had been two main traditions, shall we say, in Lisp, for some years. One was Maclisp, which was basically our stuff. The other was called InterLisp, and it had originally started at BBN, with [Danny] Bobrow and [Warren] Teitelman, and then, at some point, Bob Rowe and Tidelmann moved to Xerox Parc, when they started having Xerox Parc. And they called it Interlisp , and developed it at Xerox Parc. And at some point, ARPA put their foot down and they said, okay, let's make things common. And they convened a meeting at SRI. And they sort of knocked heads together, and we somewhat grudgingly- but, at that point, things had worked out that the two Lisp systems had pretty much evolved to the point where they had solved the major problems. In some cases, though, there were trivial differences in how certain functions were implemented or certain properties or this or that.

Anyway, there came a standardization effort, and it went through committees. There's several different umbrella organizations. Of course, if you're a company, you've got to be very careful because you can go to jail for anti-trust violations, if you're not careful with it- talking with your competitors, and one thing and another. So, we wound up forming this thing called the X-3 J-11, Standards Committee. And it was chaired initially by a guy named Bob Mathis who was from the ARPA program office or something like that. And basically it met a number of times over a number of years, and considered, in detail, zillions of different issues, and eventually hammered out something called Common Lisp, which is sort of the LISP standard, to this day. Some comments I can make on that, eventually, if we get a chance. But, anyway, essentially we had already had the early stages of Lisp machine project fairly advanced, by this time. And, so, Common Lisp wound up being Lisp Machine Lisp, I would say, about 85% or 90%. There were just a few situations where we made some accommodation to the Interlisp people, or something like that, or did something that was an improvement on both. But anyway, that committee started and kind of went on, with this bureaucracy and so on, for many years, quite a few years.

And so, basically, we made these CADRs, and it really got to be quite a hardware making operation. It was actually more hardware than any project at MIT had ever made, up to that time, and I wouldn't be surprised if even to this time because, in total, we wound up making about 35 CADRs. And we had an outstanding offer, at this time, that if you had $50,000.00 in your research contract, we would make you a CADR, which was a fantastic buy, at that time. And so there were quite a few groups around campus that took up on it. There was this plasma physics group and there were several groups over on the main campus, and so on. And there was a- particularly, a guy named Tom Callaghan. Tom Callaghan was just really the greatest guy. He was a little bit older. He was sort of in charge of our shop and did a lot of that, and stuff. He had grown up in South Boston. He knew Whitey Vulgers [ph?], and so forth, from having grown up with him. And he was just the best guy. Anyway, he sort of handled our construction operation. The actual construction was outsourced, but we actually, of course, had the parts and had to order the parts and assemble them, kit them, whatever, and send them to someplace where they got…

Hendrie: Wire wrapped.

Greenblatt: Well, wire wrap was one thing, and then stuff- boards and so forth. Then I developed a machine which buzzed out these processor panels. So, you got a backdoor of a CADR, which had six of these panels on it. And this thing had two XY probes, with probes, and it used just a plastic- a gear- plastic chains, and people said, oh, it would never- it wasn't particularly accurate. But it had a rather cute thing that calibrated itself, and it worked quite well. Anyway, so the thing could basically just sit there and go bung, bung, bung, bung, and buzz out the entire CADR processor and find any missing wires and stuff like that.

Hendrie: Wow, that's great. Okay.

Greenblatt: And so that was a good hack. And we used that and that greatly accelerated check-out of machines. After things had got well sorted, there was-it got to the point where there was about a 25% chance or so of the machine just working the first time. Of course, sometimes it didn't, and sometimes it could be very frustrating to find- to check it out.

Hendrie: Sometimes, it can take a wile to find it.

Greenblatt: Yes. And we gradually got better diagnostics, and so forth, that would accelerate that process. Anyway, so, yes. So we made quite a pile of these machines. Now, one particular contract, which was of note, and, again, we'll put in some- close some things here. But, see, in those days, and for many years, MIT had an incredibly poor computer situation, as regards its students. In other words, if you were part of a research project, then you had access to lots of stuff, where there were all kinds of research projects, et cetera. But, if you were just part of the main university, then what you had available to you was some old IBM 360 card batch thing- number one. And number two, this project called Multix, which grew out of CTSS, number one. It had a very long history, and whatever. But, basically, it was based on GE, and later Honeywell, equipment. And this was very expensive and very limited, in capability, compared to the DEC equipment. At some point, there had been a bid, and they tried to evaluate that. But DEC, it really wasn't a big enough company to be viewed as a serious competitor. And so DEC lost the bid and it went in with GE. Anyway, to make a long story short, this was a whole develop of what was supposed to be a ultility. It had a few good ideas, but it had a tremendous number- it had an incredible bureaucracy. It was based on the idea that bureaucracy is good, and what we really need to make this a business is bureaucracy. So, in order to do anything, required incredible numbers of meetings of groups and sign offs and just…. It wasn't based on displays at all. So….

Hendrie: Was it based on teletypes- ASR 33's?

Greenblatt: Actually, it was originally based on 37's, I think it was. The 35's had only uppercase. And they realized they needed lower case, as well. So there was a model of teletype called a 37, I believe- it might be a 39- that did have upper and lower case. It turned out to be a dog. So, although the system was sort of designed for that, they wound up really using Selectrics, various versions of IBM Selectrics terminals. But, the system had long names- mumbo, underline, mumbo, underline, mumbo, underline- which was actually a good thing, and innovative at the time, from a system engineering point of view. However, they didn't do anything to try to ameliorate the typing, from the user point of view. So you just had to type incredible amounts of stuff to get anything done and if one character was wrong, it would just barf at you and you'd have to type the whole thing over again. Make a long story short, the thing was a colossus white elephant, which MIT more or less forced through. Corbató and so forth, being powers that be, they managed to get some kind of an institutional commitment from MIT, which says, come what may, MIT will support this system, as a institute resource, for five years, or something like that. And so they did. But the resource was essentially very, very limited. The whole system had like 2 or 3 boxes, each of which had 128k words of memory in it. This system had a very, very large virtual address space- another one of its primary features- which was, in essence good, in some ways, in a lot of ways, but yet it was kind of ahead of its time and they didn't really work it very well. The thing is, the overhead was just very high and it was very hard to get anything done. So, this was, however, was an institute resource. It was extremely expensive. And these people had the idea that the way to make everything commercial- they had a form of the neocon [neoconservative] economics idea, that if everything is accounted for, and charge what it costs, then the forces of the market will optimize things and the system will be configured with the right amount of the various pieces, depending upon the price. And so they, of course, had to dilute this eventually. But the original idea was that you would pay for residency time, of core memory, and for transfer rates- every little operation that went on during the….

Hendrie: You'd pay for.

Greenblatt: You would individually pay for- that's right- and you would thereby optimize it by reducing your….

Hendrie: By natural economic.

Greenblatt: Yes, that's right.

Hendrie: Yes. As opposed to just getting upset.

Greenblatt: Or trying to get something done. That's right.

Hendrie: Yes, exactly.

Greenblatt: So, anyway, this project was going on this whole time. And, in fact, it was going on in our building, and much of the time the machine was on the 9th floor- they went through several generations of machines, and one thing and another. And, at some point, Honeywell bought out GE, or something, or GEI, and they moved Honeywell- yes. At some point, Bell Labs dropped out of the project and they did Unix- there's another story. And Honeywell moved in, in Text Square, in the area, actually, where this Adams thing had been partially. Anyway, so that was mostly a big rat hole, where the institutes always were going down, and very, very, very little useable resource was being devoted to students, or anything like that. So, meanwhile, however, out on the West Coast, McCarthy had this long-term commitment to time-sharing for the masses, and he was pushing that all along. And so, at some point, he evolved something called LOTS- low overhead timesharing. And this consisted of buying a 2060- off the shelf, by that time, a DEC machine, and plunking it down in the middle of a hall in the main Stanford campus. This is right next to the Stanford bookstore, just- there's some architecture, with a big internal atrium. And they just opened it up to students, on a low overhead basis. They ran the thing with just a handful of people- six or eight people which, for those days, was small numbers to run such a thing. And this was low overhead timesharing. And it was open to the Stanford students, and they got a bunch of cheap terminals and they did this and that. And it was really quite great, for its time. And MIT was being left seriously behind, in my view- and no doubt other people's. And that actually persisted, for a couple of years, that Stanford had LOTS- MIT had nothing.

Hendrie: And the name was maybe chosen with some malice.

Greenblatt: It could be. Anyway, at some point, the powers that be- very likely, Joel Moses, and I'm not sure who else- passed the hat to MIT alums, with money- namely Tectronics and E&GG [Edgerton, Germeshausen, and Grier, Inc.] folks. And they raised a fairly sizable pile of cash. And they built some new buildings- Building 36 and 38, along Vassar Street there. And, to go into the new buildings, they had some money, and so they split their money in two and they bought one of DEC- a timesharing system- and they also bought what was originally going to be half a dozen Lisp machines, at this- and we gave them an even considerably better deal. I've forgot what it was, but it's considerably less than $50,000.00 per machine.

Hendrie: These are for CADRs? Yes?

Greenblatt: Yes, these are for CADRs. And there's a little story connected with that. It was sort of even a cheapened up CADR. We tried to save some money on the disk drive and then those disk drives turned out to be losers. We had to send them back. So we would up equipping the machines with five machines, with T-80 disk drives- and one T-300- instead of all the…

Hendrie: Okay.

Greenblatt: It's a set of 6- it originally would have been six machines, one with a T-300 and five with a data disk mumble, that turned out to be a loser. Now, it turned out the Lisp machine, although it was used some, it turned out was not a great success as an instructional tool, and actually the timesharing system- by this time, DEC had TOPS-20.

Hendrie: Had a real timesharing system, that worked.

Greenblatt: Had a real timesharing system. Well, they also, over the years, eventually got TOPS-10 to a fairly presentable state.

Hendrie: To a pretty good shape too. Okay.

Greenblatt: Although it still had a number of fundamental problems. But that's right. At some point, BBN had developed a timesharing system called Tenex. Both <microphone cuts off> boxes. The original KA-10 did not have paging. It had "protect" and "relocate." And that was fundamentally not sufficient. We could go into that quite a bit. Anyway, both BBN, and ourselves, made ….

Hendrie: Built your own paging boxes.

Greenblatt: We built our own paging box and did brain surgery on the inside of the PDP-10 to install it. So, we wound up with three KA-10's, that had ITS pagers. And BBN made, I'm not sure how many things. They were a commercial company. So they were a little bit at an advantage to us. At some point, DEC realized that they were losing, sort of, and they said, okay, let's make a deal with BBN. And they made some kind of a deal with BBN where they took a bunch of BBN's people, to DEC- including this guy, Dan Murphy and they basically then worked on it for another year or so, and came out with TOPS-20.

Hendrie: Okay. Which was based on Tenex.

Greenblatt: Which was based on Tenex. And so that also had paging, and that was a good thing. Meanwhile, we also- the KL-10, however, had microcode. And so we succeeded, and I succeeded, in part, on one particular occasion. Actually this guy, Dave Moon, did the main thing. But, in fact, it's an interesting story, which you probably don't have time to go into to. I went out to DEC one time and convinced them that they should give us the source of the microcode, which was incredibly proprietary, at that time, so that we could implement- hack the paging and implement an ITS pager, in KL-10 microcode. And so we succeeded in doing that. And so our MC machine had an ITS pager- and it was a DEC machine with no hardware change, but just microcode change.

Hendrie: Very good. Very elegant. And ITS was still being used in the…

Greenblatt: Oh, yes.

Hendrie: In your lab?

Greenblatt: Yes.

Hendrie: In the- what do you call it?

<crew talk>

Greenblatt: Yes, so anyway we got the microcode. And Dave Moon eventually took that over, and Dave Moon was the main honcho for the MC machine. I just did incidental things after some time. I was involved, of course, in a Lisp machine and other things. But, we were talking about low overhead timesharing. Anyway, MIT raised this money. MIT built their buildings. MIT, the Tectronics people, in EG&G, funded this ECS computer system, which is the Lisp machines, plus a DEC 20. And then finally, MIT also had the equivalent of low overhead timesharing, and a computer for EECS students. Prior to that time, they just absolutely had nothing.

Hendrie: Wow. The Multics machines were still running at this time?

Greenblatt: The Multics machines were still running. And their prices gradually went down a little bit. And there was something called the student information processing board- or SIPB- that became quite active and that some good people were involved in and did deliver some services to students. But still, the amount of resources that students could get and what they could really do, and so forth and so on, was very minimal.

Hendrie: Nothing like the opportunities you had, even when you first got there with….

Greenblatt: Well, because I got tied up with research.

Hendrie: Because you got tied up with research.

Greenblatt: And, of course, that continued to be the case. Many students did get tied up with research groups, and they could do things. But, if they didn't- if they were just in the regular main floor of MIT courses, then they could….

Hendrie: They were really stuck.

Greenblatt: They were really out of it. And, actually, just to touch on a thing we discussed earlier, I dropped by Tom Knight's office, just a few days ago. And one of the conversations came up is that these days it's actually illegal- the Federal government being concerned about use of research money to subsidize teaching or something, has now decreed that absolutely no classrooms can go on in research buildings. So, for example, that- not only does that Dreyfus building cost a fortune, but the classrooms that it has, if any- I think there's one or something- cannot be used for an MIT class, because that would be using research money to subsidize education, or something. So it's an incredible story. And it comes up now with this biology thing. He has this thing, BioBricks [Foundation], and he's looking for MIT to try to teach his BioBricks thing. And he's at a loss, sort of, because he's not in the biology department, with access to the right kind of classrooms, really, via the normal chain of events, sort of. So here he wants to teach a biology class, but he doesn't have any place to teach it because he can't do it in the Dreyfus building, because it's illegal, and he can't do it anywhere else either.

Hendrie: He can't find another place to…

Greenblatt: This is not totally insoluble. I think eventually they made me- but it's the kind of thing that goes on. Anyway, so, that was a fairly major thing, finally, when MIT did have some sort of a student ________.

Hendrie: So now there are a bunch of these Lisp machines, out there, and students can get at them. So what happens next, in the story?

Greenblatt: Well, what happens next is, having made 30 of them, and having a pretty good sized demand, from people that are at a number of different places across the country, and it being sort of ARPA having kind of pumped up to their- to everyone's eventual regret, I guess you would say, what AI was going to do for people. It became obvious that there was going to have to be some companies. And so, I was in favor of basically spinning off a company, and having it self-financed. In other words, that we actually had quite a number of people who were willing to pay, up front, 80 grand, for a CADR, essentially. And we had a CADR production facility, involving this contracting thing and so forth.

Hendrie: Yes. You actually had a fairly good system, all tuned up.

Greenblatt: That's right. We had a system tuned up. And, so essentially we did do that. LMI [Lisp Machines, Inc.] I initially started with just a incidental $10,000.00 of my own money, and that was just printing brochures and stuff- not really very much. But then I hooked up with this Steve Wyle guy- the son of Frank Wyle, of Wyle Laboratories. And basically his father kicked in a hundred grand, I think it was. And essentially that was enough. We got a space on Blackstone Street. However, basically, [Russell] Noftsker, and eventually other people, decided this wasn't enough, and that they really should get regular venture capitalism. And so they got a guy involved, this Robert Adams, who had been at SDS, working with Max Palevesky, or whatever, and basically started throwing money around- what at the time was probably fairly small amounts, in some sense. But, in the particular situation, at that time, it was fairly large amounts. So, ultimately, we had a split. By this time, Noftsker- I had lost confidence in Noftsker.

Hendrie: Now was Noftsker originally involved in LMI at all?

Greenblatt: No. Noftsker actually had been at the AI lab and had had a falling out at the AI lab, and had left the AI lab, and had been moved to the West Coast and involved in another company for 6 years or so. But he would occasionally…

Hendrie: What was that company? Do you remember?

Greenblatt: Yes, I do. It's a company called Perceptron, or something like that. Apparently they had some kind of a manufacturing thing. And it was, quote, a success. In terms of companies, it managed to achieve income and to sort of be self-sustaining, at some level, in some ways. Anyway, I wanted to do this bootstrap thing. They didn't. And there was a number of other differences, shall we say. So, what ultimately wound up happening was, the two companies split out. One was Symbolics, the other was LMI. Both licensed the software from MIT. The majority of the original team, I'm afraid, went with Symbolics. A few people stayed at MIT, and the few people that stayed at MIT, sort of actively helped LMI- in particular RMS- Richard Stallman, who you may have heard of. Anyway, also a number of the people who were actually involved in this production of CADRs. That was one of the things, was Symbolics was planning to not deliver the CADR. Essentially what Symbolics did was take the exact design of the CADR and turn it into printed circuit cards.

Hendrie: Redesign it for higher volume commercial production.

Greenblatt: Well, supposedly.

Hendrie: Supposedly. That was their- that was what they said.

Greenblatt: That's right. And so the machine remained compatible. You could actually take a disk pack off of one and put it on the other and they were….

Hendrie: So they really were the identical design?

Greenblatt: There was an identical design, but it was- this is the LM-2 now.

Hendrie: Yes.

Greenblatt: So, that's right. So LMI made exactly the same CADR, precisely as at MIT. And we ultimately made about 35 of them, I think. We got a number of orders, on this money in advance basis.

Hendrie: Wow. I just need to interrupt, just for a second. You said an LM-2. What changes did you make between the original CADR, and when did it sort of transition?

Greenblatt: Okay. MIT had the CADR. Symbolics, when they spun off, called theirs the LM-2. This is this ________ second version of the same exact logical design.

Hendrie: Okay, I'm sorry.

Greenblatt: LMI did the same physical design, as well as logical design, and they called theirs the Lambda- let's see, the Lambda was later. <inaudible>

Hendrie: I misunderstood. I thought the LM-2 was a second version CADR that LMI.

Greenblatt: No.

Hendrie: That you were building at MIT, at some point.

Greenblatt: No.

Hendrie: No. Okay, I got it.

Greenblatt: So the LM-2 versus CADR was 100% compatible. So both companies spun off and both companies initially had some success. But then- and we did bootstrap the company. We essentially had a positive cash flow, for a period, even without any more capital. But, at some point, it became clear that we were going to need more capital. And Texas Instruments had bought some machines. And we didn't want to be building so much stuff anyway. So, it turned out that there's a company in Irvine, California, called Western Digital. And there's a guy named George White, who was an MIT engineer who was working there. He had been involved in various standardization activities, particularly something called the new bus, which later formed the basis of Apple machines, for a period. That project actually had had its genesis also at MIT, through another long history that was going on in parallel.

There's Steve Ward- another group at MIT, in the same building, on the 5th floor- had decided you should have a workstation with a [Motorola] 68000. And so they designed such a thing and they attempted to get it built, by subcontracting it out. They did it a couple of times and it flopped, because the execution of the company was bad. On about the third try, they subcontracted with this Western Digital Company- a group at Western Digital. And there was a long period where the processor of a 68000 was available, but there was no- you couldn't buy a workstation with a 68000 in it. It was quite a frustrating period, in a lot of ways, for a lot of people. Anyway, at some point, these guys developed this standard bus, which was a fairly innovative bus for its time, called the new bus. At LMI, we were saying, okay, we don't want to be in the hardware business, we don't want to make all this hardware. What would really be good is if we can buy these new bus machines from somebody, and just make our processor- or maybe not even make our processor, after awhile- and do software, and so on. So, we investigated that, and I got involved. This new bus also had a standardization committee, and I went to a number of those meetings. And, however, although Western Digital's execution was a little better, it was still not sufficient to make the grade, and they sold a few machines, but practically none. So, they were about ready to lose. Meanwhile, here was Texas Instruments. So, essentially, what wound up happening was a 3-way deal between us, Texas Instruments, and Western Digital, with the money essentially coming from Texas Instruments. And, basically, that kind of put money into this group at Western— the group was actually transferred to Texas Instruments, from Western Digital. But initially they just moved across the parking lot, to another building, in Irvine, where they started making these new bus machines, with an additional trial, and with us as a potential customer, as well as, hopefully, other people. But, as it worked out, there weren't any other people, really, or the machines weren't good enough to really make the grade that way.

So, somewhat later, there was another round of- another deal making- whereby we raised considerably more money <microphone cuts out>. We took over the manufacture of this new bus thing, and we got quite a large building out in Andover and, basically, started making new bus machines, just for ourselves, just because that's the only way we could get them. And, by this time, we also had a successor processor developed, call the Lambda. And the Lambda was a card oriented processor. So it actually did have four cards. And its main claim to fame was that it was multiprocessor, on the same bus. So it had a cache, and it had cache coherency. And, basically, you had a card cage. The processor was four cards. So you actually had room to plug in two separate processors, plus two displays, plus a memory board. And so this gave you two separate, logically independent Lisp machines that you could use…

Hendrie: In the same box.

Greenblatt: In the same box.

Hendrie: And they could talk to each other too.

Greenblatt: They could talk to each other but they really didn't, except that they- all the Lisp machines talked to each other over a local network. They all had Chaosnet or Ethernet variety, so they could talk that way. And that was good enough for the talking they needed to do, essentially. But they did share a disk drive. They shared sort of the overhead of the box. And, by this time, it became possible to make a 64-mb memory board, all in one memory board, PC, and it had SIMs kind of soldered in. They weren't plug-in SIMs, but they were kind of soldered in SIMs. And that enabled you to have 8 megabytes of memory for each of your two machines.

Hendrie: Yes. Very nice.

Greenblatt: And that worked well, and that was popular, and we sold quite a few.

Hendrie: Yes, I would think so.

Greenblatt: So, that was the Lambda, and that went on. Essentially, as part of that deal, we licensed Texas Instrument to a bunch of the software. And I went down to Texas Instrument several times, and basically brought them up to speed on a number of things. And they eventually went in, in competition, and came out with something called The Explorer Series. And we didn't really want to be in the hardware business. But, unfortunately, by this time, the company had been lost control of. In other words, the infrastructure to build this thing was this whole building full of people. Meanwhile, the control of the company was in the hands of these venture capitalists and their new presidents, who wanted to go in the hardware business- and basically he was out of control.

Hendrie: And now they had two competitors, Symbolics and Texas Instruments.

Greenblatt: That's right. So, at some point, the company got into trouble and it eventually declared bankruptcy. It also had a major other product, that it had invested in all this time and brought along, which was a process control product called Picon. And that was viewed as being a very valuable thing. And, right at the critical moment, unfortunately we got stabbed in the back. The guy who was involved in this decided to steal it- it was a guy named Lowell Hawkinson- generated a company called Gensym. Lowell Hawkinson had worked for Fredkin, actually, many years before. So, anyway, we later sued them. There was a settlement and there was a nominal payment. But, of course, by that time, there wasn't anything anybody could do, besides pay lawyers, so that wasn't…. But, anyway, that was- I don't know that the deal necessarily would have happened anyway. But there were various deals that were supposedly going to happen. And the last straw, shall we say, was that these Gensym guys decided to steal Picon. And so that caused the remaining things. There were several other players involved in the picture, at this time. There was something called MCC, which is the microcomputer consortium in Austin, who was a big user of Lisp machines, and to whom we had sold a big bunch of Lisp machines. Anyway, what happened, as far as I was concerned, by this time, was that I was pretty well tired of it anyway. But I had an employment contract. And Ward McKenzie- the guy who was the president- the first president of LMI, brought in by Venture Capitals, from TI, was a guy named Frank Spitznoggle [ph?]. The second one, who…

<Tape change>

Greenblatt: …to a situation where LMI had a large manufacturing plant in Andover in a new building. They had a research and development facility in a fancy building called a Thousand Mass Avenue in Cambridge, which is where I was basically working and the software, this Picon Group, and other software people. And the company was headed for the rocks. So— but I had an employment contract so I wanted to quit. However, what I was going to do was consult for MCC and by this point I knew the MCC people and they had a bunch of Lambdas and they wanted LAM to support and also there might have been some artificial intelligence. I know there was— what's his- the Stanford guy's name, I forgot for a moment I'll remember, anyway he was there doing his supposedly AI system and so on. So McKenzie decided that he didn't want that. He didn't want me consulting at MCC in competition with them. So we reached an agreement that well, okay if they wanted employment contracts they could just pay me. So basically for about six or eight months, maybe even a little longer, I was in a state of suspended animation where I was getting my salary but I was not actually doing anything for LMI. I had another couple of guys that I was going to form this consulting company with so I decided since I wasn't really planning to get this money I would just support them out of that salary. I had a pretty high salary. And also at 1000 Mass Ave there was some space available. And at this point since the company was contracting so much they didn't have room for all their machines so they physically had excess Lambdas and furniture and stuff. It actually was quite a big space and I just had it pretty well stuffed with Lambdas and amazingly enough there was electrical service and was able to plug them in and use them. So basically I had that space and we sort of had our little company but we couldn't consult for our major customer. But we just sort of did that.

Well, the meanwhile the company went ahead and filed for bankruptcy and a receiver got appointed and one thing led to another and then it turned out that a friend of mine, a guy named Guy Monpetit, he was a Canadian, and he was an associate of Seymour Papert and involved in the Logo children program and so forth. And basically he had become our Canadian distributor and was pursuing various schemes in Canada and so forth. Anyway he suddenly appeared on the scene with money and, to make a long story short, bought the assets from the bankruptcy court and installed me as president of a reconstituted company called GigaMos Systems and it was called GigaMos. Now, I had pretty mixed feelings about this whole thing but I decided to go along with it just feeling sort of a duty or something I don't know what. Anyway by this time a new processor had been designed, not by me, called the K-machine. It was designed primarily by a guy named Pace Willerson [ph?], who's a good engineer, and a few other people. Well, they had been working on it during this interim when this company was going through this thing and so forth...

Hendrie: They'd worked on it at LMI?

Greenblatt: Yeah. LMI, right during this thing. I wasn't particularly interested in it but then it kind of landed in my lap after this buyout and this thing. So it wound up that we moved the company down the street to 675 Mass Avenue from 1000 Mass Avenue that we kind of reconstituted a smaller company. Meanwhile the manufacturing and another large piece of company had moved from Andover to the Lowell Mill and much lower rent space but still quite a lot of it. And there was just an incredible amount of stuff. They disposed of equipment. It was just totally incredible how much furniture there was just long lines. This mill had long bays and wired cages and these bays were just filled with expensive furniture that had been bought by this company and so forth and so on.

Hendrie: With venture capital money?

Greenblatt: With venture capital money.

Hendrie: Cause it never made a profit did it?

Greenblatt: No. And so they sold all that and they sold this and that and so…

Hendrie: Hey, $40 million will buy a lot of furniture.

Greenblatt: That's right. They also bought- they bought a testing machine, a PC circuit testing machine thing for I think it was at least three quarters of a million or something. And they bought several other major things and so on. Anyway I was not only president of the entire company including that piece. The Picon guys having left we had these customers and we still had Picon and so some other people that had been kind of on the border of the Picon thing, including a guy named George Garet [ph?], and so forth sort of stepped forward and offered to sort of operate that part of the company. So that's right. So basically that happened out of this Lowell facility and meanwhile in Cambridge I worked on trying to get this K-machine processor put together. Well, basically eventually I sort of did but eventually several of the people—well the primary person was Pace Willerson with good reason he just said this company doesn't have a good environment, I'm leaving. And he left pretty early in the game. Another couple of people hung around for a while but it was a terrible working relationship and they eventually left too. And so I just kind of slogged forward and eventually did more or less get this K-machine thing going. However, at that point…

Hendrie: At what year are we maybe?

Greenblatt: Let's see, this whole LMI bankruptcy thing was in '86 I believe and…

Hendrie: So '87 or '88?

Greenblatt: Yeah and then in '88 another stroke from above, incredible thing happened, and this happened in Canada. Turns out that this Guy Monpetit had made a deal with a Japanese venture capitalist- let's see, remember his name. Anyway he was very rich from money in Japan and he was interested in Logo and children's computer language. And Monpetit had been involved in that. I had been tangentially. I mean I had been a consultant to the Logo group and involved in it but as all this other stuff happened I kind of faded off to that to a pretty low degree. But anyway, turns out that Monpetit had made this arrangement involving buying a major property in Canada, outside of Montreal that involved essentially Hoffman-LaRoche had developed a major drug research campus outside of Montreal and then Canada changed their laws or taxes or something and Hoffman-LaRoche decided not to go through with it. So here was this humungous eight story, fancy building, beautiful sitting in St. Catherine Eglise, or something 20 miles or so outside of Montreal and being completely unused. And so this deal from- [Takayuki] Tsuru was the Japanese guy's name. So anyway Monpetit also had this small silicone design company, IC design, called Silicard [ph?] that consisted of a few hackers and they had some Lisp machines and they were based in downtown Montreal. Well, by means of this deal Tsuru put through a fair pile of money such as I think $8 million or something.

Monpetit used that money to number one, buy this unused Hoffman-LaRoche property; number two, buy the assets of LMI out of bankruptcy court; and number three, also take control of a company called LCSI, Logo Computer Systems, which was a Canadian company that had— that he had been involved in earlier and that had had its own long story and was involved in this children's language Logo thing. Well, so that was a deal and that went forward. But then a totally incredible thing happened in Canada wherein some Canadian lawyers managed to get Tsuru's ear and to feed him incredibly false stories that he was being taken advantage of and basically it started a big lawsuit up there and fundamentally caused the whole thing to collapse. And they were full of stories about all this money having been taken to the US and so forth. So here was GigaMos Systems of which I was president but which I didn't own. It was a subsidiary of GigaMos Canada and there was essentially this seizer action that went on there. So basically the lawyers of Tsuru, who were tapped into incredible amounts of money, generated some kind of an international guarantee that they'd say okay, here's this LMI-GigaMos Systems we're taking control of it and we own it and we'll pay whatever expenses are involved in doing this. So it was sort of like a bankruptcy except instead of being in bankruptcy court it was actually in federal district court in Boston and it was an international thing. It involved the international guarantee that they made in order to persuade the US court to allow them to do this in effect. So you then got auditors here in Boston who had no motivation to do anything except run up a completely blank check for- and they basically did and they— and anyway so they went through the motions of trying to sell the company and doing this and that but, of course, they were just charging time at $300 and up an hour and then just incredible amounts of this and that. And no matter what they did they were tapped into this unlimited supply of money in Canada essentially as a result of that.

So essentially that was the end of GigaMos and that was about '88. There's more story— I don't know if you want to go further with this. But just to make a short thing we can say that so then what I did was I sort of went back around MIT and I saw that the fact that MIT's phone system was incredibly screwed up and I generated some technology called— eventually called Omni Phone Technology. I applied for an SBIR, a small business innovative research grant, got that and then there's a deal whereby that funds you for six months and then you apply for phase 2, got phase 2, so I got just less than a half million dollars and proceeded to develop this technology. Unfortunately again it involved standards- international standards, it involved modem things and one thing and another and ultimately it was not really successful in doing that up. But it's still something which is sort of necessary to this day and hasn't really been achieved to this day. But I got some patents on it. But anyway finally approximately at the end of that my mother who had moved to Boston in '78, much earlier, well in '90 she fell and she broke her hip and so I essentially became responsible for her care at some point about then. And then she began to have a degenerative brain disease and so she required more care. So essentially I worked in the independent researcher mode and also kind of taking care of my mother for a number of years. And then finally my mother went to a nursing home and I'm continuing to be an independent researcher.

Hendrie: Independent researcher and her caregiver in a sense too.

Greenblatt: Well, I visit her everyday in the nursing home so I'm not an official caregiver but I'm a something or other. But yeah, so that's that. And I'm working on interesting projects which is a whole other story.

Hendrie: I had a couple of questions. When LMI spun off and you started that was your partner the CEO? You clearly were the lead technical person.

Greenblatt: Yes. That's right, he was the— Steve Wyle was the CEO, that's right. I was chief technical officer.

Hendrie: And then when venture capital arrived…

Greenblatt: I continued to be chief technical officer and member of the board but, the control of the company just progressively obviously went away.

Hendrie: Where did you get your initial venture capital?

Greenblatt: Well I guess there were three major rounds altogether. First is Steve Wyle, which is sort of just initial. Then there was Steve Wyle's father plus a company called Genesis Capital involving one guy from Silicon Valley and one guy from Washington. I may have their names if they would mean anything to you although I'm not sure if I can find them right now. Let's see it was Jerry somebody. Anyway we raised, I think, a fairly small amount; I think it was $850,000 or something like that. Then there was another round where we raised from a bunch more venture capital where including new enterprise associates and a fairly long list of venture capital funds. And as part of that a guy named Frank Spitsnagel was brought in who had been a right hand man to Jerry Judkins or something at Texas Instruments— who was at Texas?

Hendrie: The NEA were the lead people?

Greenblatt: They were pretty prominent.

Hendrie: You remember who represented them?

Greenblatt: Yeah, I do actually. One guy is on Louis Rukeyser occasionally. Of course Louis Rukeyser is off- his name is Brian Rogers. And he kind of came in later. The other guy Art something— Art Marks right. Art Marks was the first guy

Hendrie: Okay, I was just curious.

Greenblatt: So yeah. At that time I think that loss was probably the biggest loss that New Enterprise Associates had had up to that point when they eventually lost their money.

Hendrie: And then you said there was one more after a while.

Greenblatt: And then there was a mezzanine thing and most of those same investors plus a bunch more. And at that point they brought in this Ward MacKenzie guy who at the time was very sought after and for whatever reason people thought that just because he'd almost replaced Ken Olsen he was- somehow knew something or whatever it was that's right. And then after- actually then another round was that just as LMI was getting in trouble and so forth it turned out that Data General wanted Ward MacKenzie and so there actually was a transaction where Data General actually paid some money into LMI for Ward MacKenzie's contract and Ward MacKenzie went off and was— I'm not sure he was, you know, chief operating officer or something. He was essentially a senior guy at Data General.

Hendrie: I sort of remember that now but I think I was long gone from Data General. I left in 1980 so it was many years later.

Greenblatt: But— and he made some attempt to hire me at that time but I was not interested in being hired. So I didn't do that.

Hendrie: And this is sort of now LMI's all ready in some trouble. Did they ever raise another— a last round or was the mezzanine sort of the last?

Greenblatt: They tried to and it was pending when this Picon/Jenson walkout occurred and that— and then that essentially sabotaged that.

Hendrie: Some of the assets walked out the door and pretty hard to raise money if you…

Greenblatt: Yeah. One of the main investors was a company called General Signal Corporation and- who was a primary player in some of this stuff. And the guy at General Signal, whose name I don't quite remember right now, was a senior guy there and their interest was Picon. So they were interested in this

Picon as being a process control thing and so forth and that was the primary part of the deal that they were interested in and so when that went away…

Hendrie: Well unless you have some more things you'd like to…

Greenblatt: I think we've pretty well covered it here and I think we're just about out of time too.

Hendrie: Thank you very much for doing this, I really appreciate this. And the Computer History Museum thanks you too.

Greenblatt: The tournament that was played April 22 to 23, 1967. So this is tournament number three.

<crew talk>

Greenblatt: And this was the first tournament at which a computer had a positive score and basically as I said it was the third one— in the first one we drew one game and lost all the rest, in the second one we won one game and lost all the rest. In this one we played only four games but won three and drew one I believe. And so this then shows the games and the analysis and I just looked over it last night again to just see what went on and what was going on here and so I did kind of play over the games and just…

Hendrie: Can I see how the analysis looks?

Greenblatt: Yeah. Here you have— here's the start of the game when the parameters are set up. And here's the portion of the game is playing in the book. So it looks like about four or five moves out of the book. And then once it's out of the book each page here shows the position, the list of legal moves and their plausibility, and then the results of the analysis. This number here is the evaluation. Zero is an even game, positive is good for white, negative is good for black. And so…

Hendrie: These are evaluations?

Greenblatt: Well those are cutoffs. This minus the thing— this- the machine is playing white so it's trying to find the most positive number here. And so in this case the most plausible move is the one it's going to play. And here is the analysis. You know it's going to play bishop to queen night 5, it's expecting bishop to queen 1, bishop takes bishop, knight takes bishop. And then it- having looked at this line it then goes back to that position and searches a little bit deeper in this thing where it says okay, well after that if we do another small search from there we get pawn takes pawn, knight takes knight.

Hendrie: Could you hold that? Could you just go over the first two pages so I can be point— I'm just going to pick up the camera. Do you have time to do that, just for a second?

Greenblatt: Well I have time but I don't know how much you're going— I mean we can get out a chessboard here and get into this position and stuff.

Hendrie: I think it's interesting how you read this because I'm going to ask you about whether you'd be willing to…

Greenblatt: Here's a list of legal moves with their plausibility. So these are moves which are scored and then this is the analysis. So it starts with the most plausible move, it expects this reply, this reply and this reply. That's step four. In this case there's no further search deeper necessary then. Then it comes back to the top level and says okay, well that's interesting, if we go to this end position and do another short search what do we get. And so it goes there and gets two more plies here. This number here is the time and seconds so this whole move— it will print here. Well it looks like it doesn't. Sometimes it prints a total- yeah, there's the time, 54 seconds. So at this level I think you had about two minutes per move.

Hendrie: Okay, you had two minutes to work on it.

Greenblatt: Something at a fraction but something like that. So anyway that's one move. The machine plays bishop to queen knight 5 check and then the opponent replies bishop to queen 2 and here's the next move. Start all over again. The machine eventually won this game so let's see we get to the end here. Let's see we're playing white, move 15, okay, here's our tag. And yeah, you see here at the end the numbers are getting substantially positive for white which means that white's winning or seems to be winning so that's good. So that's a game and there's a total of four games.

END OF INTERVIEW