BOB EVERETT LECTURE MARCH 15, 1990
DIGITAL'S HISTORICAL COLLECTION PROGRAM
LECTURE TRANSCRIPT

KEN OLSEN: When I went to work at the computer lab at MIT almost 40 years ago, it was a fascinating experience. It was a little bit like going into a religious order. It was a military project; no one knew about it on the campus. I really wanted to work on the numerically controlled milling machine computers. I didn't even know about them. They ran out of money for those, just those few days when I was looking for a job, and I accepted one at the computer lab. The environment was strange because the job was strange. Making a 10,000 tube computer when the design life of the vacuum tube was 500 hours, you took a different approach to things. The technology used 1/10th microsecond pulses, which is very fast for those days. The only time people had done anything close to that was in radio frequencies, and shipping pulses around was still a strange idea. So all the technology was new.

A number of ideas that came out of that laboratory were the basis for Digital. One of them was that that environment, which had almost the feeling of a religious order, was that passion for reliability and discipline. Some of the things may not have been necessary, but the overwhelming desire for reliability because the job was impossible and the discipline that went with it. Besides that, was the feeling of freedom, free communications. Risk taking, experimenting on things you wanted to experiment with, talking to anybody, complaining about anything. Intellectual encouragement and challenge. Tying those two ideas together was one of the basis for Digital. Freedom, try things, experiment, and yet extreme discipline.

One of the other ideas, probably [a] very significant idea, was the idea of interactive computing. There was only two or three machines being built at the time, maybe four or five. All of them were six months for completion. This was the only one that was interactive. The generosity at MIT, the openness, was something we've always tried to capture and sometimes have failed on. They would allow certain students to come in, this military project, and use the computer at night. It was interactive with the cathode ray tube. They were so generous they either purposely, or didn't know it, allowed them to change the read only memory, It was soldering diodes, until one day they put the diode back in the wrong place for the morning.

Bob Everett played a key part in this. I have some parts from the earlier generations of Whirlwind. It was supposed to be an analog machine, then a serial machine, then a parallel machine had to be fast. I think Bob was the one who, as far as I could tell, (he would never say so himself), did the logic design. It was an approach to making computers which for many years afterward was strange. We got involved...involved is an understatement...with IBM. As a result of that there were two groups at IBM, the traditional computer approach and the Whirlwind approach. With their help, and I like to think with Digital's missionary effect, we changed the whole world to the MIT approach, which I think was Bob Everett's approach. What he didn't do with Boolean algebra, maybe he'll tell us. My immediate boss was Norm Taylor, who is joining us today; his son works for us in California. He was my immediate boss for many years. When I walked in the door, you know, I was a young kid, out of the Navy where I was absolutely fascinated with the radar with 150 tubes. Here was this thing with 10,000. I found the smartest guy in the whole place, and sat down beside him, every chance I had to learn. That was Dick Best. Dick Best would spend time drawing circuits so that they were easy to understand about how you drew them. [That's] something that's influenced Digital ever since then, trying to make things easy to understand. The cathode ray tube we had, there was no amplifiers. They didn't have amplifiers for cathode ray tubes for years. You went directly into the plate. I learned from Dick how you could measure within a small fraction of a volt something with no amplifier.

The fascination of those times will always be romantic.

Everybody played a key part. Bob Everett was the associate director, Jay Forrester was the director. Jay was a very bashful but yet flamboyant leader, full of ideas. Bob was the brains behind the outfit, always sober, always right. He did much of the design. [They] made a wonderful team. Nobody could compete with us. With that team going after them, boy, we never lost. It's a great privilege now to introduce Bob, who for several years has been our director [on the Digial Board of Directors]. He only made one mistake in life that I know of. [LAUGHTER] We invited him to be our Chief Financial Officer at one time and he turned us down. He ended up heading MITRE Corporation which the country needed badly, but we never quite forgave him for that.

[APPLAUSE]

EVERETT: Thank you, Ken, for that flattering introduction. I'm really worried about the possibility I'll begin to believe what you say after awhile. But I appreciate it. I think Jay and I were a good team. It was a great time. In fact, it wasn't just Jay and me, it was all of us and I think it was that feeling of a team, of this common, open, everybody can talk about everything, problems were everybody's problem. I think that was an extremely important part of it. As a social organization I think it's something that you can't beat.

I sometimes wonder if it's partly I was younger that it was a wonderful time. Speaking of being older, I've gotten a little bit used to talking about history but to find that I'm billed as talking about pre-history is [LAUGHTER] going back a long way.

Ken and I started at MIT. I'm older than he is; I got there first. I got there in 1942 to go to graduate school, and went to work in the Servomechanisms Laboratory, for Jay. We spent the next couple of years building a stabilized radar mount for ships, very different from building a computer but a very chastening experience for an engineer. We finally got it into production at Westinghouse Air Brake. The production was ramping up -- I think that's a good Digital term that I've learned recently -- and we had to do something else, and we got connected to the special devices center of the Office of Naval Research (which was probably not the Office of Naval Research at the time). This was during the war. The Special Devices Center was run by a man named Louis deFlores -- a remarkable individual -- who among other things, had become famous for inventing the cracking process for gasoline. He ran this outfit which built simulators and training devices. They're really some rather marvelous little things done with the technology of that day. One of the things they built, probably the most complex, was an airplane simulator which used a realistic cockpit from an existing airplane and then this was backed up by a computer which ran all the dials and knobs and switches and things of that sort and enabled you to sit in this airplane and more or less fly it. The computer was made out of servos, thyrotrons, and relays and they were wonderful; they'd click and whirr and flash. Computers today are so dull, but they were really great in those days. They didn't work very well but they sure made a fuss about it. Anyway, [CHUCKLES] Louis, and others, had the idea that instead of building a thing like this which was then tuned up to look or feel like a real airplane, if they built one that solved the equations of motion and the aerodynamic equations of an airplane, in a general sense, and then they set the parameters based on analysis or on wind tunnel tests, you could then make an simulation that felt like a real airplane which hadn't been built yet. This would be a very powerful tool for airplane designing.

I've learned since from talking to aerodynamicists that there was a great deal of disbelief about this in the aerodynamics group and around the country, but it was an interesting time. Nowadays, you probably couldn't get to do it, because all these people would pop up and say it's terrible and then there would be investigatory groups and studies and whatnot, but back in those days nothing like that happened and if deFlores thought it was a good idea he gave us some money, and we thought it was a great idea and we started to build it. As Ken said, it was originally intended to be an analog machine built with servos and ball disc integrators and all sorts of things, and it became obvious in short order that that was a very difficult thing to do. The dynamic range required of the variables in order to really simulate the airplane were reasonable envelopes, very difficult to get mechanically. But we built a lot of pieces, were working away at it and a classmate of Jay's, Perry Crawford, who is now at IBM and was at that time at the Special Devices Center, introduced Jay to the electronic digital computer. After a certain amount of thought it became evident that that was a very new and interesting and powerful device which looked like it would solve the problems that we were after and enable us to make a computer which had the flexibility, dynamic range and so on that was needed. So we dropped the analog computer and started building the electronic digital computer in 1945. After awhile we became convinced that there were a lot of applications more important than the airplane stability control analyzer, and there were changes in the Navy. Louis left, and the Navy lost interest in the simulator and we never actually built it. But the computer survived, and became Whirlwind.

Now I mentioned all this for a reason -- to talk about the situation as it was back in 1946. The war was over and there was a lot of interest, in both the United States and Great Britain, in electronic digital computers and there were a number of groups -- I don't remember how many but there must have been upwards of a half a dozen, mostly at universities -- who were engaged in building electronic digital computers. This came for a number of reasons. There were some good new ideas like stored programs; they were coming out primarily the Moore school, von Neumann, and Eckert and Mauchly. There was the availability of technology and pulse circuits and in storage devices which came out of the war primarily that were developed for the use in the radar people. There were groups of capable and experienced people who were becoming available who were used to building things during the war and were excited about that and wanted to continue to do so. Not least important there were the mechanisms in place and funds in place for supporting things of this sort. In other words there was a great opportunity which a lot of people saw but there were also the people on the resources and the technology and the momentum left over from the war that made it possible for all these groups to work on this digital computer. If today one would be forced to write a requirement and point out all the things it would do, and how you were sure that you could do it, and things of that sort, and it would make it very difficult. Things were different back in those days, I think much more promising as far as developing new things are concerned.

Most of these other people however, as Ken said, were interested in a different kind of computing. Essentially all the groups were driven by the need to solve some problem they had. Most of the problems came out of the universities, and they had to do with the solution of equationis which described natural phenomena of one sort and another. A lot of the demand was for the solution of partial differential equations; some people just for investigatory or scientific purposesl; other people to build nuclear weapons; still others with the hopes of predicting the weather. There were some people who were interested in business accounting -- making out the payroll and things of that sort -- that had grown out of the IBM punched card equipment, which was widely used for that purpose. The only group that I'm aware of, anywhere, that was interested in the machine and the possible use of an electronic digital computer for control purposes, was our group. Now after all, we'd spent the war working on control devices -- this was the Servomechanisms Laboratory -- we had a project to build a large simulator. We'd come out of the engineering tradition and were used to engineering discipline while the other groups were primarily led by scientists and mathematicians.

I think the group at MIT had some other advantages as well. We were accustomed to large complex projects. I speak now, not for myself so much, as for MIT in general. During the war it had run the radiation laboratory and built a lot of interesting things in other laboratories. The general idea of building complicated things was not foreign to the group. We had, therefore, the understanding and backing of the MIT leadership, a very powerful force, especially with all of the reputation that the scientific and technical community had as we came out of the war. Another thing that we shouldn't forget is that as part of MIT we had access to a steady flow of very capable people out of the graduate school. After the war there were a lot of these people, in many cases who hadn't acquired degrees before the war, [but who] had gone into the war and served as radar or communications engineers and gained a lot of experience about how the world worked and come back to go to graduate school using the GI bill. A lot of them were married. MIT had very strict rules so there was a very highly select group that came; these people needed research assistantships and needed to get attached to laboratories. We used to pay them as much as $150 a month. We were very attractive because young people liked these things. Here we were building this marvelous machine and we used to get the cream of the crop. It used to make the professors very mad. This was really a tremendous source of people; they came in and they went through the graduate school and we were able to keep a large fraction of the very good ones as permanent staff members. Ken and lots of other people came out of that and they were a tremendous source of strength to the organization.

The result of all these things is that we were interested in a different kind of computing for a different kind of purpose. Realtime systems, like simulators, like control devices, have realtime things in them, like people: you can't turn the time scale of a person up and down. So if you had a system that involved both a person and a computer, the computer had to keep up. If it was unable to keep up, you couldn't just stop things for awhile while it did: it really had to keep up. Secondly, it had to be very reliable. If it broke down all the time in the middle of this test, or this experiment, or worst still, when you're controlling something expensive and complicated, it's can be very unsatisfactory. As far as speeds are concerned Whirlwind ran at about .04 MIPS. That seems utterly trivial today, but it was a lot in 1950. Most of the machines these groups were working were on the order of a few KPS. Reliability was a special concern to us, too. Everybody was worried about reliability because, as Ken says, if you've got 10,000 tubes and they last 500 hours each, you're going to spend all your time fixing the machine. There were many people who felt that you could never build an electronic digital computer out of vacuum tubes and get any useful work out of it because you'd have to work on it all the time to keep it running. The upshot is that everybody paid attention to reliability. But the people involved in non-realtime activities had a lesser demand on them because if you ran a program and it didn't run through well you could run it over again, or if you had a very long program you could stop once in a while and store the intermediate results. So the upshot was that if you had a 50/50 chance of getting your problem done and if the machine was available maybe half the time when you wanted it, you could use it, you wouldn't like it, you'd be annoyed at it, but you could use it. So people were careful but they had a more cavalier attitude toward it, relatively speaking, than we did. In our particular case that was just unacceptable. The machine, in our opinion, had to run without failure for several hours most of the time that you asked it to do that, you know, better than 90 percent. That's for simulation and so on. For the actual control of important things it had to do better than that. And like today's special computers it really should run all the time.

I'll say a little bit more about the SAGE Direction

Center which was built in the '50s by us and many others. But the spec for that, -- and we wrote the spec -- was that the Center should not be down for more than four hours a year. It had 55,000 vacuum tubes in it and the requirement [for it to] run all the time, [with] no unscheduled outage more than four hours. It was done with a duplex machine, and it was done, as Ken says, with complete attention to reliability, every decision put reliability first. Now if you look at military electronics today, you'll find a lot of it doesn't work very well. A major reason that it doesn't work very well is that the people who design it do not put reliability first; it's not because they can't or they won't, its [that] their customer, the military, asks for performance first. In fact, the conversation usually takes the form of, "I want this thing to be reliable, you got that? Okay, now let me tell you what I really want." The other kind of thing beside a computer that really works well is a satellite, and in my opinion, satellites really work well for the same reason that computers really work well, and that is people thought you couldn't build them at all and that meant that the history and discipline of the thing was to put reliability first. Satellites put reliability first, even more than computers do, because once you've fired that thing that's the last chance you get. The SAGE Center, as I say, had a duplex computer, it was a sort of a poor man's Cirrus, if you can use the phrase "poor man" in regard to a SAGE Center, which probably costs on the order of $250 million a piece in today's dollar.

Those were the operating requirements on us and it's why things were the way they were when Ken arrived. We really did pay attention to everything, there was a great deal of discipline, setting of standards and setting of the way you tested things. For example, nobody was allowed to solder anything in the machine unless he had been through the soldering course. I never went through the soldering course and I never was allowed to solder anything -- I wouldn't have touched it under any circumstances, anyway. I'll give you an example of one of the differences between our group and one of the other groups. The concern with computers was not so much permanent failures because you can find permanent failures and fix them. The concern was with intermittance. So if you had an intermittent in a solder joint, then you may have a problem. Our solution to that was to be very careful to wrap the leads around the pins, to solder them twice, to inspect them three times. This other group decided the way to do it was to lay the pin, the lead across another lead, and put a solder in between and put some tension on this, on the theory that if the joint started to go bad it would pull open and then you could find it.

Now let me show you a few pictures of our machine. This is the Barta Building where we all worked back in the '40s and early '50s. It's the old Barta Press Building down at MIT right near the campus. It's still there and used for other purposes. I might say that one of the big advantages of this building is that it's over the railroad from MIT and therefore did not fall under purview of buildings and power and we were able to keep the building clean ourselves and so on and fix it and change it without having to go through the bureaucracy down at the Institute. That's another lesson that we all got...[LAUGHTER]... I don't know anything about the situation at Digital Equipment today so I won't comment.

The computer occupied about one-third of the top floor of this building and eventually with all the other things that were added to it, it occupied pretty close to half the building. As I said, we spent a great deal of time worrying about reliability and vacuum tubes came first. Vacuum tubes fail for a number of reasons, one is a sort of infant mortality that you can get rid of by burning them end first for awhile and that gets rid of the early ones. Secondly, they tend to sort of gradually deteriorate. We found that one of the reasons was that the vacuum tube manufacturers put some things in the nickle of the cathode to make it easier to handle and this would gradually grow an interface between the metal and the oxide coating causing a resistance and the tube would gradually deteriorate. Others mechanical problems just resulted after all these tubes were poured off the line at about 30 cents apiece when we arrived and the solution to this was very, very careful. The tubes were specially made for the laboratory. They cost on the order of $10 apiece which today would be about $100, which I think is in the order of a million times what a transistor costs on a chip these days. One of the things that we did was a thing called marginal checking, invented by Jay. The circuits were very carefully designed to have wide margins and we found that if you properly selected the voltage, like the screen voltage or something of the sort, and you could move the operating point within the margin and essentially measure the margin. So you could pick out gradually deteriorating components, in particular, vacuum tubes. This is a five digit multiplier which was an attempt to build a small piece of Whirlwind, try it out, and see what kind of reliability we were getting and how the margin checking was working. It multiplied two five-bit numbers together and checked the answer. The young fellow in the saddle shoes is with us today, that's Norm Taylor. You're not wearing your saddle shoes today! This worked very well, in fact at one stage of the game it worked for 40 days without making a mistake.

There were only a few groups; you could get all the people working on computers in the world together in one room if you wanted to, and very often did. People came around and visited each other all the time, disagreed with each other on almost every subject, but they came and talked to each other. [Howard] Aiken [from Harvard University] came to see us and Jay showed him the five digit multiplier and how it worked 40 days and Aiken, I don't know how many of you know Aiken, but he was a big impressive looking fellow, he shook his head and on the way out he said, "Well, five bits that's about one decimal digit.....goes as the square of the word length, ten decimal digit's a pretty good word, so that's a factor of a hundred, so that's about a half a day, that's not bad." And off he went. [LAUGHTER] But 40 days was good.

This shows you how the tubes came out finally. At the time this measurement was made in 1950 there were 3500 tubes in this record keeping -- and we took very detailed records. This was the number of failures. There was about 2500 hours, so there's about 10 million tube hours or something in this list, and there were 18 failures not due to tube characteristics, that were not found by marginal. So that's 18 out of ten million hours. That's a lot better than 500 hours but that's the kind of numbers you need in order to do what we were trying to do.

Another thing that happened, and Ken mentioned it, was how we used to connect things right to the scopes. Now I'm eternally impressed, as I wander around the laboratories, by the beautiful instrumentation we have these days, that we didn't have that in the early days. We had to build most of the stuff we wanted. This is a test facility, and if memory serves me, that's a diode switch on the top. One of the things we found is that there was hardly anybody working on the computer, everybody was working on the test equipment. This led to a decision to build standardized test equipment, standard building blocks, the things that were mentioned earlier that you could plug together and get any pulse strain you wanted and gates and things of that sort. This was done, and built in large quantities, and made available to people, so if you wanted to put together a test set, you could get most of what you needed out of the stock room instead of starting over.

This is a typical element, a pulse generator. Eventually Ken built a series of these things made out of transistors which we used and then, if I understand correctly, that was your first product at Digital.

People built a lot of things out of these; people have built whole computers out of them on occasion.

The machine was spread out in two dimensions so that you could get at every piece of it. This is one digit of the accumulator. It contains two flip flops and a number of gates and it's about this wide and about this high [DEMONSTRATING]. We were very careful about putting a flip flop in the machine. As Ken said, we went to a lot of trouble to design the circuits properly, to have standard circuits, to have standard pulses, to have standard connections, so that you could plug things together with reasonable confidence they would work. Then the logical structure, Ken says we treated it like a puzzle, but I thought we treated it like an engineer would, it was like designing a bridge or something, simple and easy to understand and easy to check out. And if you start being very complex in those days it would have been very, very hard first of all to have any confidence that it worked with reasonable margins and secondly, to find out what was the matter with it if it didn't work. This kind of thing was relatively easy to find out.

This is a picture of the early days of the computer and it was a big computer; it filled a room which is probably not an awful lot smaller than this one. It was quite high. One of these columns is a digit column. In the arithmetic element, it would have the particular flip flops that went with the particular registers in the arithmetic element. This is the test storage. We were very worried about the storage tubes which we intended to use, we were developing them ourselves. Storage was always the biggest problem in the early days of computers. We decided we put in what we called a test storage, which was 32 registers of toggle switches, plus five registers of flip flops which could be inserted into any of the places you wanted. That was a solid confidence test storage which you could use for setting things up and testing things. One of the problems we had with it is that there were 32 times 16 plus some extras, there were over 500 switches you had to set to set up a program and I'm not sure anybody ever succeeded in doing it right the first time! One of the reasons being that we had designed it like you see here. This is on one side of the aisle, so if you look at it this way, the biggest digit is to the left and the smallest digit is to the right, but the switches are on the other side so that the [LAUGHTER]...the words were all backwards and that made it hard. But by then it was too late to change it when we found that out.

This is the control on the left. It was designed for flexibility. There's a 32 way switch which set up horizontal lines and then there were vertical lines went to the gates which could be fed from the various pulses sequences. You could solder in diodes and change the instructions. I assume that's what Ken meant when he said someone was doing that in the middle of the night; if someone was doing that in the middle of the night, he wasn't supposed to do that [LAUGHTER]. Nobody was supposed to do anything to the machine without putting a red tag on it and filling out the log. But it was easy, and there was a temptation. If somebody ran out of space he'd say, "Well, if I just change this instruction a little bit, you know, I'd get a little more room..." There was a lot of temptation to do things like that.

This is the control room. That's Jack Gilmore on the left; he works for Digital now. That's Joe Thompson, who as far as I know, is the world's first computer operator. The machine brought out all of the switches and lights and things so that you could get at them. The display is on the upper right there. This is a young lady checking out a paper tape, we used paper tape input and Ferranti tape readers. These are the magnetic tapes; we used magnetic tape drives we bought from Raytheon. We were willing to buy anything we could get from anybody that was a good thing to buy; unfortunately there wasn't too much of it. This was mechanically good, electrically it wasn't very good. The trouble in those days with magnetic tapes was that there would be flaws in the tape, motes of dust or something, which would push the tape away from the head and you'd get a dropout. These are six channels wide and we discovered by testing it that there was almost never an extra bit. I don't recall ever getting an extra one. The dropouts were not across the tape, they never exceeded more than about two channels wide. So by ganging the channels three apart we were able to get one marker channel and two data channels and the thing worked very reliably. As evidence of what you could do with 32 registers, to test this out we wrote a program which ran the tape unit which recorded blocks of (I think) 128 ones across the tape, then went back and read them all. If there was a dropout, it would stop, print out the block number and a picture on the tape of the magnetic tape itself. That was done with 32 instructions. It's shows you what you can do with if that's all you've got.

This is a sketch of the tubes that the Digital Computer Laboratory built. The gun on top wrote on mica screen and then there was a holding gun; so the idea was that this was a permanent storage, you didn't have to go around and refresh it. We succeeded in making those and we succeeded in making them work. It proves what you can do if you have to, not that it was a really good thing. We finally got two banks of 32 tubes total in operation and a tube construction facility that was able to make tubes fast enough to replace the ones that were wearing out. It did work but it was a very difficult thing to manage.

This shows Jay Forrester holding an example of the tube. Pat Yutes, the head of the tube shop, and Steve Dodd, the head of the storage tube facility and one of the tubes under construction on the trolley. This shows the tubes in the rack. They were put into a case and trimmed up so they all looked alike, theoretically, and then they were put in, in pairs, in each digit column with the necessary amplifiers and gate generators and so on.

One of the things that, as far as I know, we did first, was to connect a display to a computer. This was done by taking two of the flip flop registers in the test storage, putting digital analog converters on them and driving the XY plates of a CRT, and then the computer would intensify when it needed to. Eventually, when we put many of these on the machine, it was so arranged that all the tubes were deflected at the same time, but the computer would just tell which one was supposed to do it, so you could get different displays on all the different tubes. We did this first because, I guess, we were the first people to have this kind of problem. The issue of communications between the individual and the machine on a realtime basis required tubes and required other kinds of things. For instance, here's a light gun. There are lots of switches, but the computer would display on the display tubes say, tracks, or whatever you wanted, and then you could tell the machine to do something by putting the light gun (the photo cell) over the spot you're interested in. When it intensified that spot, it would get a response, and that would tell it was that one you were wanting to work on. We also built joysticks, and cursor movers...nobody ever invented a mouse, that I can recall.

By about 1951, Whirlwind was operating with one bank of storage tubes -- that's about one kilobyte -- and we were starting to get involved in a different application. Before we had the simulator, we had done a lot of studies of things for the Navy and we'd done some work on air traffic control, but the Navy had changed. We were now under the Office of Naval Research and it was used to giving $50,000 grants to mathematics professors. Whirlwind was spending the appalling sum of a million dollars a year in developing the machine, and they finally decided that was too much. So they were going to cut us back to where we just could use the machine but no longer add to it. But as so often happens, at almost exactly this time, a new application arose and that was air defense. The Russians had aircraft that could reach the United States on one-way missions. They had set off an atomic bomb. People really worried about it, it was a high priority thing. The possibility of using electronic digital computers for handling radar data came up and the Air Force took over the support of the project and went so far as to create a new laboratory at MIT, which is the Lincoln Laboratory. Most of the Digital Computer Laboratory became part of the Lincoln Laboratory, although there was a residue which stayed at the Institute. It was the first machine on which many people actually had the chance to experiment with an electronic digital computer, and it did things like working on computing tapes for the first digitally controlled milling machines and a lot of other applications. It was the machine which Lanning(?) and Zeiller(?) used in their first algebraic compiler.

So the laboratory was formed, and in '52, there were really two major decisions. One of them was to build an experimental sector; this is the experimental sector known as Cape Cod Those are the locations of the radars which were connected to it. Eventually there was a big radar at Brunswick and another one down at Montauk and one at South Truro on the Cape. The others were small radars, gap fillers for looking for low altitude aircraft. All of this was brought together in Whirlwind and used to make a model air defense system.

That meant adding a lot of things. One of the first things we came up against, and we keep coming up against this and other people do as well, is software. Prior to this people didn't seem to worry too much about software. If a fellow wanted to solve a problem, he sat down and wrote the program. They were fairly short, they couldn't be very long -- there wasn't any place to put a very long one. It turned out that a capable fellow -- and they were all engineers, there were no computer scientists, the software was initially done by engineers -- a good engineer could write a program on, say 1,000 instructions over a weekend and check it out and make it work. But this problem required, say 20,000 instructions. That turned out to be a different matter. They were all done in machine language. They were all done in address and it required now setting up standards, finding ways for people to work together, making what would now be called CASE tools and operating systems and things of that sort. There was a lot of that going on.

One of the things that was needed was a drum memory. There were really two drums. One was a secondary memory which was capable of holding these 20,000 instructions plus all the data, and which could be put into the central machine. We bought the drum from ERA in Minnesota. We also used a drum for a buffer; we had to get all this information in. We didn't use interrupt, we didn't have the time to; if you started asking the machine to stop what it's doing and handle interrupts it would have been a serious problem. So we used buffers. They were kind of input/output boxes. The data would come in from the phone lines, be stored on the drum with markers, and when the machine got around to it, it would read off all the data that was marked, clear the markers and leave them open for something else. So those two drums are in the bottom corner here along with the electronics.

This is some of the other stuff that we had to build for all of the displays and switches and input lines and everything else that was necessary. This is one of the operational rooms. These show actual Air Force operators sitting at consoles, running the consoles, initiating tracks and directing interceptions and things of that sort. The man who was in charge of this, Bob Weiser, Ken knows well. I told him that I thought that was a pretty messy looking thing, it looked to me like it was a warehouse full of displays and he never forgave me for that because he said it was very carefully worked out and I'm sure it was. [LAUGHTER]

The thing I think that really saved the whole business was the core memory. Jay had thought of the core memory sometime before, and at a very low level, one or two people at been working on measuring core characteristics and so on. It seemed obvious that maybe we could make Whirlwind work with storage tubes, but the idea of making SAGE work with storage tubes would really boggle the mind. So we really put a lot of effort into the core memory. Ken made large contributions to that. We started out with wound ribbon cores, and after awhile we got some good ceramic cores, and eventually built a 32 x 32 or 1K word memory out of these cores, hand wound by young ladies. That's a core plane out of Whirlwind.

This is MTC, [MEMORY TEST COMPUTER] you recognize it Ken? When we were working on the storage tubes, we had a test facility in the basement where there was a lot of this pulse digital test equipment which would generate various patterns and so on for testing the tubes. Then we subsequently put the tubes in the computer and could use the computer for testing them. I became convinced that the best possible test device was a computer so when we started to build the memory I told Jay that we needed a computer to test it. My recollection, which may be wrong, is that he was very reluctant. He said, "You guys just want to build a computer," [LAUGHTER] and I denied that. I said we really needed it. So finally he agreed,

with the proviso that it couldn't multiply [LAUGHTER] and you didn't need to multiply to test storage tubes. So I told Ken that we could build it, but it couldn't multiply and he went off and built it. This is well known as Ken Olsen's first computer. He's made hundreds of thousands of them since, but this is the first one. One day when we'd got the core memory working and everything worked fine, I mentioned to Jay that this was all a great success and how about letting us multiply. He said sure. I called up Ken and I said, "It's okay to let it multiply." I think it was three and a half seconds afterward [LAUGHTER] that it officially multiplied.
It turned out to be a very useful thing. We used it for years thereafter. Essentially all the terminal equipment input and output gear and so on for SAGE was tested using this computer. It's worked so well we made a second stack so we had 2,000 words or 4 kilobyte of memory, wonderful. We put it in Whirlwind -- it was all done in a hurry -- and it had the most profound effect. It doubled the operating speed, as I recall, and it quadrupled the rate at which you could bring information in from the outside. It changed the mean time to failure on the storage row from something like two hours to two weeks, and it cut the maintenance time from four hours a day to two hours a week, and it freed up the whole tube shop to work on display tubes. It was like a miracle for the whole system.

This is something about Whirlwind's performance; the 13,000 tubes included the extras to make a directions center out of it. It required an hour a day of maintenance, had an MTBF about ten hours and about 96-1/2 percent available. This doesn't mean that we didn't have problems with it. We used the Cape Cod system not only for experimentation and design purposes but also for sales purposes. There was a steady flow of visitors into the laboratory. We used to run an exercise a week and invite visitors. Weiser called it the little theater off Central Square. You get a whole room full of generals and the machine was supposed to work! Whirlwind had a personality all of its own, it was sometimes fractious. It turned out, for some reason that none of us could figure, that Whirlwind really liked George Valley, who was the associate director. If George was around, it would work like a charm. So if it was important to us to get a good demonstration, we'd always get George to come and then the machine would work fine. He would go out the door and it would collapse. It might stay flat on its back for days afterward.

Another major decision in '52 was to get on with the problem of getting computers built. This meant the design of a new computer intended for the air defense problem and also getting some organization to build them.

We went through a source selection in 1952 and picked IBM. The number of people that were capable of building computers in the United States was very small and IBM seemed to be outstanding. So we contracted with them. The first contract was in '52 and the first prototype was delivered in late '54 to MIT. Everybody underestimated the resources required. IBM kept adding people; the Air Force kept adding money, and they ended up eventually building a plant to build these in Kingston, New York. We originally intended to design the machine ourselves and turn it over to somebody to make lots of them. It didn't really work out that way. MIT had limited the size of the laboratory, we couldn't add anybody, there were about 180 or so staff members that in Division Six, a lot of whom were involved in other things, so there were less than 100 engineers available and it just couldn't hack the job. IBM ended up with hundreds of people. My major recollection of those days is not of making design decisions but of talking people into turning jobs over to somebody else and so that they could be made available to take on the new things. There was a steady flood of new problems coming in the door and we somehow had to kick the old problems out in order to get the time. We spent our time looking for people, for computer time, for space, for help, trying to stay ahead, trying to stay on schedule. We missed the schedule by one year, which is, I guess, pretty good by today's standards. It made us feel bad, but it was one year. It took six years from the decision on the part of the Air Force to build this thing until we had the first operational center turned over to the Air Force. We had a lot going for us. The big thing we had going for us was that the Air Force gave us its full backing; they were committed. They told us to go ahead and do what we had to do and if we needed money, they gave us the money. There were lots organizations who were available, lots of resources and a lot of effort.

One of the things that amazes me about it now as I look back on it is that the [average age of the people involved] was less than 30. An old man was 35! Here were all these billions of dollars being spent and all of these high powered industrial organizations doing what we told them. I think there are several reasons for it. In the first place, the group was competent and did have a good track record. We worked very hard to persuade people of what we wanted, and not just order them to do it and take their opinions into account. But one of the most important things was that nobody else was willing to take responsibility for it. A lot of people said it couldn't be done, that it didn't make sense, and so nobody would stand up and say, "Well, I don't agree with you. I'll take responsibility for doing it differently." Everybody said, "Well okay, you're responsible, we'll do it your way." Nobody wanted to hold things up because it had such a high priority and they didn't want to take responsibility; the only other alternative was to do what somebody said, and they did what we said.

We also had a lot of interesting problems with software. It reared its head again. The SAGE operational program, as I remember, had about 100,000 lines of code and there were about 500,000 lines of support programs of one sort or another, compilers and editors and operating systems and test programs and data reduction programs. This was a truly enormous amount of code for those days. And it was in fact the software that caused the slip. The Lincoln software people made what I believe was a wise, but difficult, decision, and that is that instead of just starting in to write 100,000 lines of code and hope for the best, they first made a substantial investment in support tools, I guess now would be called CASE tools, before they wrote the ops code. that meant that the actual writing of the code was late, but it also meant that the code worked when they got it written. That's good trade if you have to make it.

The plan was to prepare and maintain one master program, not let people in each center adjust the program. Then the master program had to be adapted to the particular geographic weapons and so on in each center. That took a lot more people than we had expected. The result was we needed to recruit a software organization because there were about 50 or 60 people at Lincoln, and that wasn't anywhere near enough. Now there were no software organizations back in mid 1950s and none of the industrial corporations involved were interested in doing it. We tried to talk IBM into it, but they said they built hardware and it was up to their customers to set up their problems. We said "You're making a mistake, look at the future, you're going to need all these people." They said, "Why don't you run your university and we'll run our company." [AUDIENCE CHUCKLING] They had said that to us on other occasions, too. That was a mistake on their part, clearly, as they recognized afterward. SAGE gave a tremendous boost to IBM not only in the total revenue and the profit they made but I think more importantly in the technology and the support for R&D and the facilities and the very large numbers of trained people they got for doing all kinds of things, from computer design, maintenance, system design, field service, all kinds of things. They just got a tremendous boost upward and they could have had several thousand software people, too; but maybe it's just as well they didn't, Ken.

Eventually we got a division of the Rand Corporation to do it. Rand spun them off to form the System Development Corporation, which started out as a non-profit then turned to a profit-maker. It was sold to Burroughs and is now part of Unisys. We went out and hired literally thousands of people -- schoolteachers, mathematicians, God knows what -- because there were no people who knew anything about software. They were trained to be SAGE programmers and put to work, shipping around the country to these various installations and things of that sort.

[END OF TAPE 1]

BE: We had to work, of course, on communications. In order to build this thing at all, we had to send digits over telephone lines from radars to radars from all kinds of things. There was some initial work that went on at the Cambridge Research Center which was transferred to Lincoln. Then we ended up not only doing that but providing computer-to-computer communications. Computers would be within central organizations in between many facilities of all different kinds. Once again this was all new. Once again we tried to get the telephone company to become interested but they said they couldn't imagine why anybody would want to send a pulse over a telephone line, and if we insisted on using their telephone lines for such purposes, it was on our head. In fact, the first time we got a telephone line from the telephone company to connect some digits into Whirlwind, they insisted on giving us a handset. We told them we didn't want the handset. They said they were sorry, but telephone lines come with handsets and we had to to put it on the shelf. I don't know if they ever got it back or not.

[LAUGHTER]

After awhile they came to understand the importance of this, and dived in and really did a good job. Now it became obvious to us that SAGE was going to be connected to a large number of places, and if we had to make them each special cases and solve the problem between communicating with them, we weren't going to get anywhere. So we did solve this problem by setting up a communication standard, and told everybody if they wanted to talk to SAGE they'd have to use the standard. That worked quite well except for the Army. The Army and the Air Force were having a fight over whether the Air Force could tell the Army what targets to use. Until that matter got settled, it was impossible to settle the technical problem of how to connect to the army's missile master. Eventually they ironed it out and the engineers didn't have any particular trouble solving the problem, until it came time to hook it together, then wouldn't you know, it turned out one of them was Big Indian and the other was Little Indian. [LAUGHTER] So Murphy was working full time back in those days too.

Another problem we discovered was the computer base system is not something you build then leave alone. Everything changes all the time. There has to be some organization that looks after it. It took about a year to persuade the Air Force of this. It took the Air Force about a year to decide that they didn't know of any way to solve it except to ask MIT to do it. So MIT said they wouldn't do but it would in fact spin off the part of Lincoln that was working on SAGE; and that turned into the MITRE Corporation. Before I go on any further, I should show you a few pictures of SAGE, for those of you who haven't seen it. That's just a diagrahm of a small number of the things that were hooked into the central computer. There's a SAGE center, that big block. The building behind it is the power system. The one behind that is the cooling tower. It's all made out of concrete. It looks like Uncle Scrooge's money bin. That's one half of the arithmetic element for one of the duplex computers. It's much more dense then Whirlwind but it's still not very dense. That's the control room. As I say, there were duplex computers, one on either side plus control for all the intermediate stuff in the far end. That's one of the control rooms. There were about ninety people who access to the machine, controllers of one sort and another. This was all done with about a tenth of a mip. In looking at air defense systems that have been built since -- systems with lower demands on them then this -- they are frequently built with five or ten mips. So my question was, what happened to the factor of fifty? I think the major thing is Parkinsonian. I think the system will use whatever number of mips somebody provides. We only had a tenth and we used a tenth. If we'd had five, I'm sure we'd have used five.

I didn't mention that we continued to do research on computers while SAGE was going on. We did discover that various things came up in the SAGE activity. They swamped the group that was working on them. It was evident that if we didn't do anything about it, we'd come out the other end with SAGE, but with no organization, no R&D program. The solution to this was to put aside a group to work on that on further-out things; tha [was done] with an eye to the future since we weren't sure what we were going to do afterward but we wanted to do something. I think the group did some very good research, built a transistor-driven core memory and built TX-0 and built TX-2. It built a sixty five thousand word core memory which was needed in SAGE.

We knew about transistors when the SAGE design started and it seemed to us that they were definitely the wave of the future but they weren't here yet and it was not possible to build a machine with transistors as they were then and meet the schedule. I think that was probably a wise decision. In fact, a couple of years more and we'd probably we'd probably gone for transistors.

By the middle of the mid-fifties, this organization in which we were all so proud, began to come apart. Jay decided he would go back to the Institute. He felt that he'd done all the important things as far as building computers and air defense systems, and he went back to the Institute as a professor and invented the field of system dynamics. A number of people left and started various companies. The most successful alumnus we have, of course, is Ken and the Digital Equipment Corporation. He took with him some of the people and some of the ideas and some of the lessons we all learned but in my opinion, the best thing he took with him was himself. He's the best electronic engineer I've ever known. I think that all of us who have ever known him have looked at the success of Digital with amazement, and we're all impressed, but I don't think anybody that knows Ken is all that surprised. Some of the software people went to SDC. Most of the system engineering people went to MITRE Corporation. The R&D part of the organization stayed at Lincoln and eventually evaporated. It kind of exploded and spread all over, and left its mark in many different places. Thank you very much.

[APPLAUSE]

KEN OLSEN: I'll make two comments. I was canoeing in northern Canada. We came out in an Indian village in Hudson Bay. There was a well built airport with a big hangar and docking facilities. You could see the remnants of a large housing facility, didn't quite realize what it was until we flew away. A tiny Indian village. That was one of the radar sites. I should have known better. The massiveness of that air defense system is unbelievable. There are twenty-three, I think, of those block houses. When they tore down the last one, we sent people up there to look at it, and take some of the pieces down to the Museum [Computer Museum, Boston] Gordon Bell [ex-VP, Engineering, Digital] calculated casually that the reliability of that 60,000 tube system was equal to the integrated circuit computers we were building. It even had people on the top of the John Hancock building, the Empire State building with binoculars feeding into the system. There were two sweeps of radar across Canada.

Another thing: Bob was right, it did IBM a lot of good. But from one of the young snots who fought tooth and nail with IBM all the time, I will always say that the project success was due to the management at MIT and the management of IBM. From my point of view, IBM really went into that for the good of the country. It was their motivation, it was risk, no obvious payback whatsoever and they deserved everything they got from it. The management at MIT was something I was really proud of and we'll forever admire their group. Thank you all.

[APPLAUSE]

[END OF TAPE 2, SIDE 1]