

BILL GATES

ORAL HISTORY

**COMPUTERWORLD HONORS PROGRAM
INTERNATIONAL ARCHIVES**

**Transcript of a Video History Interview with
Bill Gates
Co-Founder, Microsoft**

**Recipient of the 1993 Price Waterhouse Leadership Award for
Lifetime Achievement**

Interviewer: David K. Allison (DKA)
Curator, Division of Information Technology and Society, National
Museum of American History, Smithsonian Institution

Date: April 1995

Location: Bellevue, WA

DKA: We thought we'd start talking about your family. Tell me some of the qualities that you attribute to your family that later shaped your career at Microsoft.

BG: My dad was a lawyer and my mom was very involved in business activities as a board member in non-profit organizations like running United Way Campaigns. She was the Director of the University of Washington, banks, that kind of thing. They shared what they were doing out in the world with my older sister and I as we were growing up. So, we always had a sense of, "Okay, this is the Governor coming to dinner, or here is this political campaign, let's get involved in this." I was a page down in the State Capitol of Olympia, Washington. Then I went out and spent some time being a page back in Washington, DC. I understood about contracts and things. I was interested in the business world, reading about it all the time. Sort of always playing around with the idea of "What would I end up eventually doing?"

DKA: Were other of your contemporaries equally interested in the business, or did you find yourself unusual among the groups?

BG: Well, when I went to Lakeside School, I was about 12 years old. I started there in seventh grade. That was kind of a change for me. It is a private boys' school. Very strict. At first I really didn't like the environment. I did eventually find some friends there, some of who had the same sort of interest, like reading business magazines and Fortune. We were always creating funny company names and having people send us their product literature[laughs]. Trying to think about how business worked. And in particular, looking at computer companies and what was going on with them.

DKA: What companies, in particular, did you like to follow?

BG: The first computer we used was a GE time-sharing system. It was connected over a phone line. Actually, the school couldn't afford a full phone line, so someone in the offices had a switch where you could take over the phone line. It was an ASR-33 Teletype with paper tape connected up to a GE computer. But, very quickly, we found out about PDP-8s, and eventually got one loaned to us. And then eventually, Data General Nova got loaned to us. So, it these companies making smaller computers that were very fascinating to us. Joining all the user groups. DEC had one called the DECUS User Group. Getting on every mailing list. In Datamation they had these bingo cards where you could check everything you were interested in. So, we just put our name down and checked everything in there and tried to learn about the world of computing.

DKA: How did the faculty respond to your interest outside of your curriculum compared to your interest in your own studies?

BG: Well, I was relieved from some classes, Math in particular, because I'd read ahead. So, I had quite a bit of free time. So, when the Mother's Club, which did this rummage sale, got the money for this Teletype and a certain amount of time to buy computer time, it was a question of who was going to figure this thing out? Now, I was very young. I was in eighth grade and some of the older students kind of barged in and thought they could figure it out. And very quickly, the teachers were intimidated. So, it was sort of a group of students reading the manuals and trying things out.

You would type the programs off-line on this yellow paper tape and then put it into the tape reader, dial up the computer, and very quickly feed in the paper tape and run your program. They charged you not only for the connect time, but also for storage units and CPU time. So, if you had a program that had a loop in it of some type you could spend a lot of money very quickly. And so we went through the money that the Mothers Club had given very rapidly. It was a little awkward for the teachers, because it was just students sitting there and zoom! -- the money was gone.

I wrote a tic-tac-toe program and a couple of other base conversion programs. It was the BASIC language running on this GE system. So, people didn't know what to think because teachers were fairly dignified in those days and usually were supposed to know what was going on. They were okay about it, but then when the money was gone, they had to start billing us for all of our usage. We had these kind of funny student checking accounts so my friends and I still stayed very active. We were kind of desperate to get free computer time one way or another.

The amount of time we'd spend in this particular room that had the Teletype was quite extreme. We sort of took over the room, myself and two other people. They called it the "Teletype Room". We were always coming up with schemes to get free computer time, and eventually did with a local company. Convinced them that because they had a deal with DEC for this big computer, an early PDP-10, serial number 36, that if they could find problems with it they wouldn't have to pay their rent. Having a few of the students, including me, bang on it and try to find bugs seemed like a good idea. And particularly, let us do that mostly at night. So, we were going down to this ... it was called Computer Center Corporation, C-Cubed, in the University District, staffed by some old people from the University of Washington Academic Computing Center had gone over there.

So, for a few years that is where I spent my time. I'd skip out on athletics and go down to this computer center. We were moving ahead very rapidly: BASIC, FORTRAN, LISP, PDP-10 machine language, digging out the operating system listings from the trash and studying those. Really not just banging away to find bugs like monkeys[laughs], but actually studying the code to see what was wrong. The teachers thought we were quite unusual. And pretty quickly there were four of us who got more addicted, more involved, and understood it better than the others. And those were myself, Paul Allen, who later founded Microsoft with me, Ric Weiland, who actually worked at Microsoft in the early days, and Kent Evans, who was my closest friend, and most my age, was killed in a mountain climbing accident when I was in 11th grade in high school. So, the four of us became the Lakeside Programming Group. We were the hard-core users.

DKA: Clearly, your extracurricular activity was probably more important in your later development than what you did in class, or at least equally important [Bill laughs]. How do you feel as we look at problems in education, that students that are coming up interested in technical things should balance those. Do you think it is important to explore things on your own at that age?

BG: Self-exploration is great, because you develop a sense of self-confidence and an identity of "Hey, I know this pretty well. I know this better than the teachers. Let me try and see if I can understand at the next level. Maybe I'm pretty good at this stuff." And particularly with the computer where if your program is wrong, you know you try it and if it doesn't work and then you fix it and try it again. It is kind of a feedback loop, which because the classroom has a lot of people, and maybe there is not a subject that you think you are good at or interested at. It is just fascinating to try and figure out the computer.

I remember at Computer Center Corporation they had hired in some of the great people of early computer days, including Bob Gruen, Dick Russell, a guy named Weir who was a Stanford guy. Anyway, these guys would kind of loan us deep manuals on the system, just for a few hours and then take them away. So, we'd spend those few hours just reading carefully. It was so exciting to get a little glimpse and beginning to figure out how computers were built, and why they were expensive. I certainly think that having some dimension, when you're young, that you feel a mastery of, versus the other people around you is a very positive thing. And for me that came in several ways: the reading I was doing non-computer related; in Math. But computers, timewise, for many years was a key center of excitement.

DKA: You mentioned the PDP-10 as a computer that was important to you. In what ways do you think working with that particular machine set you up for the kind of developments that you did later?

BG: DEC always named their computers PDP, Programmable Data Processor. It is actually funny, because they kind of skip numbers. PDP-1 was their first. And it is kind of strange, I did a lot of work on the PDP-1, but not until I got to Harvard, late in my career. They had so few computers there, a couple of really decrepit PDP-1s hanging around were the best graphics display machines. Anyway, the PDP-10 which whose predecessor was the PDP-6 very similar and later called the PDP-20 or DEC 20, and was, in a sense, the first time-sharing system. It was written, well, it was the first...I don't know how they said that? It was DEC's first time-sharing system, in any case. And it was a very good time-sharing system. That means that you could have lots of people working at the same time sharing the storages and the CPU.

It is kind of ludicrous today because the speed of that machine is so much less than these PCs. [Bill gestures to PCs set up in interview room] To think that it was so expensive that we all had to share it and how slow things were. I mean, connecting up to these computers ... these Teletypes, the print head only moved at 110 baud, which is about ten characters per second. There was no lower case. And one day at this Computer Center they brought in a GE Terminate terminal. It was an absolute, incredible breakthrough, because it printed 30 characters per second. It was just incredible! [Bill imitates sounds of Teletype] We'd all just sit there and watch it. And even more fantastic was that it had lower case letters. So, you could do real correspondence and things like that.

PDP-10 today is considered kind of a strange machine because it was 36-bit with these 18-bit addresses. You used octal instead of instead of hex because having a word size that breaks into 18 by 18 that's not multiples of four. Anyway, DEC always used hex, because PDP-1 from the beginning was actually an 18-bit machine, although their later machines of great success were the PDP-11 was 16-bit and VAX and its successors were 32-bit machines. In any case, the PDP-10 was an exciting machine to be involved with because a lot of university campuses were getting these things and doing time-sharing. It was being used in a lot of new applications. And DEC was very much the up and coming company in computers. People were doing great peripherals to hook-up to these things. They were doing some awfully good system software. The BASIC they did which we got a hold of the BASIC source code and enhanced t quite a bit was great. They had AID and FORTRAN. They didn't have a very good COBOL until a little bit later on. But it was a very solid time-sharing system. Lots of great experimentation going on with it.

DKA: One last thing I wanted to ask you before we talk about the Altair. In addition to having this intense interest in computing, you maintained an interest in business in this early period at Lakeside and actually began a business. Would you talk a little bit about Traf-O-Data and some of the other entrepreneurial activities that you did?

BG: Well, there was the interest in the technology and also the business together. The two sort of went together. When we wanted to get free computer time, you had to have a sense of how you could motivate people to give you free computer time. With C-Cubed, that meant just finding bugs for them. Eventually, DEC made them pay for the machine. But we had been deemed useful enough by them, and although they didn't pay us, they gave us free time. That company went bankrupt. Then we found a company down in Portland, Oregon with the same PDP-10 that let us write a COBOL program, a payroll program, a huge complex payroll program. I learned about labor reports, taxes, and all sorts of mundane things. We called this group the "Lakeside Programming Group." And Paul really got interested in the machine. And Ric Weiland got interested in writing an Editor. So the two younger members of the team ended up doing most of the work on this payroll program.

In fact, they thought there wasn't enough work to go around, so they kicked me off. I said, "Look, if you want me to come back you have to let me be in charge. But this is a dangerous thing, because if you put me in charge this time, I'm going to want to be in charge forever after." Which in high school, when you are two years behind, it was considered a reasonably uppity thing. So, they brought me back in. And we got the payroll program done. We got a lot of free computer time. During all this time we were helping this school with its scheduling. It's kind of a long story, but they got it all screwed up. They used an approach that we had not recommended. And, eventually, they had to turn it over to us. So to be a student and being determining who's in what classes and when they meet. The school paid us pretty well. We actually used the computer time we had to do that. That worked out great. We bought some equipment from a bankrupt computer company, some DEC tapes and made money on that [laughs].

Traf-O-Data was taking road volume data and converting it into reports where you have actually just a 16-channel paper tape on the side of the road. The pressure sensitive hose that you drive over has a counter in there clicking out a count every five, ten, or fifteen minutes. Those have to be processed for the State Road Departments, to give out money for repairs, and decide how to do traffic lights, etc. Anyway, it was data that needed to be processed. So, we got involved in that.

We were truly naive in that we built a machine using the very first decent microprocessor, the 8008. I funded that development and wrote some of the software. And Paul Allen and another hardware person, Paul Gilbert, who knew how to do crazy wire wrapping, took these very tiny Intel memory chips -- I forget if they were 256-bit or 1K-bit, -- I think 1K-bit Intel chips in this 8008 and created a system and we did special software. 8008 wasn't capable of running a BASIC Interpreter. That had always been my Holy Grail because I had played around writing a little bit of a BASIC Interpreter on a PDP-8, and a Data General machine. But the 8008 is basically an 8-bit machine with no programmable stack. Doing this traffic analysis software was pushing the limits. Anyway, we were making machines and we thought we'd sell them to people. But, when the guy from the County that Seattle's in came to see it, it didn't work. We ended up being okay successful, not seriously successful [laughs], just by processing the tapes. At first that was a very manual process. Then we used this prototype machine that we built to do that. So, we made a little bit of money and had some fun with it.

I was definitely always pushing us out into those frontiers of "What should we do?" Even so, I never had this image that I'd end up being in computers. I didn't think computer people were mainstream or very interesting in certain ways. So, I took my first job where I took off part of my senior year in high school and worked for TRW, who was doing an awesome project using PDP-10s. Even then I was thinking, "Okay I'm going to go to college and do a normal career." In fact, it was very exciting when I was taking this time off, I got my college acceptances. So, I could just work hard on this. This was a real-time data project, controlling the power grid in the Northwest using PDP-10s. It had reliability requirements way beyond what the PDP-10 could deliver. So, we were really trying to push the state-of-the-art -- really pushing the state-of-the-art. It never met the full power industry requirements, but they did put the system on-line. And even at that point we'd thought, "Wow, this microprocessor is going to do something incredible.

As early as 1971, Paul and I had talked about the microprocessor. And it was really his insight that because of semi-conductor improvements, things would just keep getting better. I said to him, "Oh, exponential phenomena is pretty rare, pretty dramatic. Are you serious about this? Because this means, in effect, we can think of computing as free." It is a gross exaggeration, but it is probably the easiest way to understand what it means to cut cost like that. And Paul was quite convinced of that. So I would sort of say to Paul, "Well, you know what that means?" And he'd say, "Yeah, that is what it means." It is kind of fun to know this, and think, gosh, how are companies going to react, how are they going to respond to something that phenomenal? The early days were very slow moving, though. By the time I went to Harvard, all there was the 8008 chip. And the 8080 was just coming out, which was the first good general purpose microprocessor chip that Intel was coming out with.

DKA: We are going to talk a little bit about your transition into Harvard before you got to the Altair. You mentioned that even though you worked in computing, that it wasn't your goal when you went to college. Do you want to say what your goal was?

BG: That's right. I finished up at TRW, went back, and graduated from Lakeside School. I picked, out of the schools I'd been accepted to, to go back to Harvard. They seemed to have a lot of different things. I knew that if I wanted to be a Lawyer or a Mathematician, Harvard had good courses for these things. Once I got there, I thought Economics was pretty interesting. And I felt that I understood computers well enough, that I really didn't need to hang out with a computer crowd there, because they weren't as interesting. I did end up taking a few computer courses. But most of what I did was not related to computers. And meanwhile, Paul Allen, who had worked with me on everything, basically[chuckles], and who shared this idea that we should go do a company -- he actually tried to convince me after TRW that we should start a company then, and try to build PDP-8-like systems around 8080 chips. But it just was too vague and my parents wanted me to go back to school.

So, I went back there. But Paul was there and we were always talking about, "Could we stick a lot of microprocessors together to do something powerful? Could we do a 360 emulator using micro controllers? Could we do a time-sharing system where lots of people could dial-in and get consumer information?" A lot of different ideas.

But a major milestone for us was when we were walking through Harvard Square, one time, and saw this Popular Electronics magazine. And it was kind of in a way, good news and bad news. Here was someone making a computer around this chip in exactly the way that Paul had talked to me, and we'd thought about what kind of software could be done for it, and it was happening without us. And for all we knew maybe they had some software people, they were just going to go charge off and do this thing.

So, we wrote this company immediately. Sent them a letter. We have a copy of that somewhere. But anyway, a letter on Traf-O-Data's stationery, because it was the only stationery we had. We offered to do a BASIC for them. And they thought that was interesting. They called back and said, "Well, you're serious? We have a lot of strange people calling us."

This article received immense interest. I mean the idea of a kit computer -- even though there was really nothing you could do with it. There was no Teletype hook-up in the early days, there was no software for it. All you could do was use these switches, key things in into this front panel and maybe do a little program that does things in the lights. Or, actually, a guy name Steve Dompier discovered it, because this bus is unterminated, if you are very clever about the program you run, you can get high-frequency emission that can cause a radio to make interesting noises.

Now, eventually, we did get controllers for Teletypes and cassette tapes and floppy disks, that kind of thing. But, in the early days it was pretty useless. People just bought it thinking that it would be neat to build a computer.

DKA: I want to hone in on this idea of your proposing that you would actually write BASIC for this machine. It is kind of a gutsy thing to do. Did you feel about it as being boastful, or did you really think you could deliver? Did you think they'd take you up on the offer?

BG: Well, I had no doubt that I could write a BASIC Interpreter. I thought through in my head all the things that I hadn't done before on those mini-computers. Doing things really small, fascinated me. These machines, you could actually buy it with no memory card. After we got there a lot of people were calling up asking, "What do they do with it?" Well, if you don't buy a memory card you can't do anything. But, the memory cards were 1K. So, if you bought four of those you could have a 4K machine. 4K bytes. And I thought I could do a pretty decent BASIC. Fitting the BASIC in, allowing you to have both your program and your data storage in 4K bytes. That was going to be hard. But, it was a fun challenge.

People had done 8K BASICs on the PDP-8. But this was a much better instruction set. I had some ideas on how to do things a little bit in a new way. There was no doubt in my mind we could write a BASIC. I was fairly self-confident in those days. We didn't know how long it would take us. And it was kind of funny because we were sort of acting like we had it already. We went to work day and night. Paul first worked on the simulation software. He took the Macro 10 Assembler and defined macro, so we could just type in sort of a form of 8080 code. Then he modified the DDT-10, the symbolic debugger that was on the 10, to understand these instructions. He then wrote a simulator to simulate these instructions.

It was a reasonably simple instruction set. Paul was very good with the PDP-10 Assembler. I, in the meantime, laid out the design and charged off coding the BASIC. Paul later came in and helped out with that. A third person, Monte Davidoff, sat down for lunch with us and said he knew floating point packages. So, we had him write some of the math routines. And then we just kept squeezing it. So, we wrote without ever seeing this machine [pats the Altair], except in this picture, and the simulator and got the BASIC running. And then we called them back and said, "By the way, what's the sub-routines for reading a character from the Teletype and writing to the Teletype -- how do you do that?" And we got Bill Yates on the phone, who was the co-author of this article. And he said, "Well, that is pretty interesting. All these other guys call us up and say they are going to do things. But nobody ever asked us how you get data in and out of the machine. You guys sound pretty serious. You ought to come out and show it to us." And, because we'd never had the chip, just the book from Intel, if we had made any mistake in terms of how the instructions worked, the thing never would have run.

And so Paul was scheduled to fly out to Albuquerque. He decided to go get some sleep. I stayed up all night reading the book to see if we'd miscoded some of the instructions. And finally, decided it was all okay, punch out the paper tape, and made sure Paul got that before he went off on his plane. He wrote the bootstrap loader, that is the thing you have to key in to make this computer smart enough to know to go get data off the Teletype to read it into memory. He wrote that on the plane on the way out. It was actually 46 bytes, the first one. I eventually wrote in 17 bytes.

He took the BASIC to MITS. They had a machine they had run with 6K of memory, which for them was a big, big thing. And loaded up the paper tape. The first time, for some reason it didn't work. The second time they loaded it in and it worked. Of course, the simulator is very slow because you go through lots of instructions to a single instruction. So, actually, the real machine [pats the Altair], even though it is such a simple little microprocessor, was faster than our PDP-10 simulator. About five times faster. And so to Paul, when it finally came up and said, OK ... Actually that first version said READY. Most BASICs when they are ready, say READY. Later when I was squeezing bytes out, I thought, well it is faster to print OK, and it is kind of a nice friendly word. So I shortened it to OK a little later.

Anyway, it came up, it said Ready, and he typed in a program, Print 2 + 2, and it worked. He had it print out squares, sums and things like that. He and Roberts, the head of this company, sat there and they were amazed that this thing worked. Paul was amazed that our part had worked, and Ed was amazed that his hardware worked, and here it was doing something even useful. And Paul called me up and it was very, very exciting.

Pretty quickly we decided that we ought to get out there and really help these guys get their act together. I never became an employee. Paul was their Vice President of Software. But I moved out and whatever I did from the inside, I did on behalf of Microsoft. I got out there and a lot of what I started doing at first was actually enhancing the BASIC. 537

DKA: Let me ask this Bill. You mentioned that, even before this, you and Paul had had many discussions about the future. How did this work affect what you thought the future was going to hold?

BG: Well, Paul had talked about the microprocessor and where that would go and so we had formulated this idea that everybody would have kind of a computer as a tool somehow. Not just for business, but also for something they would play around with as a home device. We knew that however it got started, that there would be certain standards built-up around it, about how you programmed things. We wanted to be part of that excitement.

And so we saw this machine as just the beginning of an era. And this company was a wild company. I mean they were actually bankrupt before they did this because they had gotten screwed up doing Kit Calculators which had been their thing they had done after model rocketry.

MITS actually stands for "Micro Instrumentation Telemetry Systems", funny little things you stick on top of the rocket that tells you what the temperature is at the top of the flight or eventually, they had ones that would take pictures. So, they had done okay in that and then got into Kit Calculators. But was wiped out by Bomar and TI. And then just as a desperate thing, they did Kit Computers. When these computers came out at \$360, the price of the 8080 chip was \$360. So people kept saying, "They must be broken chips, it must be fake." And, of course, when they put these kits together, they didn't preassemble them, so if you miss one part -- a lot of people had a hard time putting these things together.

But, a lot of people got it done and eventually went on to buy the Teletype and BASIC, and actually get a running system. So we thought, "Hey, are we really on to something here? We think so." And MITS was just great because it was just a center of activity for those first few years. We went around the country in this big van, big blue van, they had, with these machines starting up user groups and demonstrating things. Actually, before we even shipped BASIC, somebody stole the demo copy out of the van and started copying it around and sending it to different computer clubs. There was a real phenomenon taking place there, right around this Altair computer. In fact, the MITS guys were kind of upset when people would imitate this computer, same plug-in bus for peripherals -- things like that. They really weren't sure what to do about it.

DKA: How did your development on this lead you and Paul to setup your own company as opposed to just working for MITS?

BG: Well, we knew that MITS was only one company, and we wanted our software to be used on all the machines. And even the original deal we did with MITS talked about our ability to get paid for licenses to other companies. We saw that as Intel was promoting their chip to intelligent terminal makers and other computer makers, that instead of that company writing their own BASIC, we could sell our own BASIC to them for a lot less money. In fact, just do it on a royalty basis. And so we went around convincing companies like NCR and a lot of the terminal companies that they should make microprocessor-based machines.

So I was never part of MITS. Paul worked there for about eighteen months and he hired someone else in to take over their software department. And I started hiring other people. In fact, before Paul came back, we had started BASIC for 6800s, 6502s -- other chips. We started the FORTRAN compiler.

We were talking with Texas Instruments about doing a BASIC for them. It was kind of complicated at first because, actually, I went back to Harvard for a little bit, and we hired in Ric Weiland, one of our high school friends, and Marc McDonald, another high school friend to sort of be the core group until both Paul and I got there full time.

DKA: So, you were convinced that you should setup your company to deal with a variety of different computer vendors?

BG: We had Microsoft as a separate company. In a certain sense, if things hadn't worked out, I could always go back to school. I was officially on leave. I didn't have a family to feed or anything. But I was doing the payroll, writing the taxes, doing the contracts, figuring out how to price the software. In fact, I was business-oriented enough that I wrote a letter about software piracy, sort of complaining that a lot of these computer groups weren't paying for their software. That really became a *cause celebre* at the time: "Is it fair that this guy is asking for money? Should we pay for this stuff?" MITS was very controversial because some of the memory boards they had been shipping didn't work. They'd been late with a lot of things. So, some people felt like it was a way of getting back at MITS to take the BASIC.

We had the first computer convention. Their people came in. So, Microsoft was a business from the beginning. Not that we had any clear view that it would ever be a large business, but I had to pay these friends that I had hired. At a minimum, I had to make enough money to write their paycheck, and if I got enough confidence that we could sell a lot more, then I would be able to hire even more people to get ahead, to be the leader in doing lots of products that could share code with each other, and take the market.

DKA: Before we move on, there is something you might want to show us, for the record, on the terminal there to get a sense of what BASIC was, and how it was used?

BG: Okay, this Altair is actually a running Altair. And what we've done is we've loaded the BASIC into its memory. So, this has an 8080 chip in it and 16K of memory. So, at this time, this would have been a big machine, 16K. It is connected through a serial card that is in the bus here. This bus is an S100 bus. Up to this Lear Signal terminal. These were fairly cheap terminals of the day. But you can tell that it's not really very dense because this was before they had cursor keys on these terminals. And so this is the manual for this BASIC.

We did different versions of BASIC. We did a 4K BASIC that was the most modest. We did an 8K BASIC. We did an Extended BASIC. This is actually Extended BASIC here. Later the most advanced version we did was a Disk BASIC, which is what you see here as the early floppy disks -- this is actually a hard-sector disk -- these little holes you see there. That is a sector marker. This is how IBM originally did these floppy disks. These did not work well at all when they first came out. You couldn't take a disk from one machine and run it in another.

But the thing that I wanted to show you -- we've actually got it running here. I can list out this little program I typed in and run that. That is my little sign wave, printing program. One of the nice things about this BASIC is it has this so called direct mode. So you can PRINT 2 + 2. It prints the square root of ten. By now, we had things like this edit command, where you can say, EDIT, like and sit there and step through and change what is in the line.

This BASIC, although it was very much modeled on what had been at Dartmouth, which was what I encountered when I used that first GE time-sharing machine. We'd gone beyond it in a number of ways to let people add machine language sub-routines. Even new words in the language like PEEK and POKE to let you read and write memory locations, were starting to come up here. I'm a big believer in interpreted languages, not only from the beginning of computing, but the future of computing. It was really the right approach, because you could just type the thing in and immediately see what was happening. And yet you could add new capabilities very easily.

Another machine in that time was this IMSAI machine. In fact, that was the next company to come out. It was sort of the first kind of clone-thing going on in the PC world, because they used the same bus that this computer here the Altair used. And so the same add-in cards would go in. So we were able to license our BASIC over to this company. When MITS got that kind of competition, they actually went back and looked at their contract and were kind of disappointed to see that they hadn't tied up exclusive rights. In fact, they made it very hard for us to license BASIC to other people. And there was an arbitrated dispute where they were told that they were completely wrong to have done that; and in fact, had their license terminated. They were bought by a peripheral manufacturer called Perdec, and actually ended up disappearing fairly quickly, as did this company here. Most of our early customers were out of business very early on.

DKA: Bill, let's talk a little bit about the people that formed the early Microsoft.

BG: Paul and I were the founders. During the time we were in Albuquerque, which was 1975 to 1978, we ended up with about sixteen people. This is a picture we took towards the end of that time [Bill refers to the group photo taken in Albuquerque], myself and Paul. This is Gordon Letwin, who had worked and did the Heath BASIC, Benton/Harver BASIC. Then he was upset when they were licensing my BASIC. So, he came to work for us and did some incredible work. Marc McDonald was actually our first employee. Other than Andrea, who wrote the manuals, and Marla, who helped keep the books.

I was the Sales Department, Contract Department. Everybody else here were programmers. We all wrote an immense amount of code. These were exciting years. The number of new machines coming out were pretty dramatic.

Our offices were here in fancy Albuquerque, up on the eighth floor of this building here. Albuquerque was great. There weren't many distractions there, but it was hard to recruit people as we tried to grow.

DKA: What was the culture of the company like?

BG: Well, I would program sort of night and day, and tell people, "Hey, we promised this thing to be done in a few months we've got to get these things done." We were so aggressive at just getting things done. Like committing to write a ROM BASIC that would fit into 12K of ROM. It was fun because, I at that time, got to look over all the code that people did, and talk to people about where we would go with things. It was just a very small group, and yet between the new machines being done in Japan and the U.S., every week something new was happening.

DKA: Did the other people share your intention? Were they there as many hours as you were?

BG: Well, certainly most of the people did. And we were all quite young. I may have set the most extreme example. But, the work was really fun. We always had deadlines that we ended up committing to that ended up being very challenging.

A great example of that was the work that we did on this TRS-80. The Altair BASIC comes out in 1975. In the next big wave is a set of three machines that came out in 1977. The TRS-80, the Apple II that we have over here that came out, actually, without the disks at first, and a machine called the Commodore PET. And those were low-cost and yet, they weren't kits or anything. They came out prepackaged. And they looked like they would really ignite the volume in the market. And all three of them went out and did very well.

Several of them, when they first shipped, had BASICs that the company themselves did. This had a Level I BASIC, that actually Steve Leininger enhanced off of Lee Chin(?) Wang's Tiny BASIC. But they knew that it was pretty inadequate. And so they licensed the BASIC from us that was built into all these machines thereafter that they called Level II BASIC. We even left some hooks in there so we could sell a BASIC called Level III BASIC, that went even further.

So, Radio Shack, with its distribution and its name set the market on fire. Apple, because they really went out to computer dealers and did a good job, far better than people like the MITS guys and the IMSAI guys. They really thought of this as a market where they had to develop the channel and do new things. And the Commodore machine, the PET, was actually the most aggressively-priced machine. It had some very innovative things. These machines drove the market and eventually, a year after they were out, all of them had our BASIC built-in. So you could even move programs back and forth between these machines because of the compatibility that we had built in there.

DKA: You mentioned that often they started with their own BASIC and then they came to you. What was it that made your products stand out and made them come to you?

BG: Well, our BASIC was fairly deep. The BASIC they first put in this machine was really, really limited. It just wasn't going to be expandable. And they wanted to put on graphics. They wanted to put on a disk. They wanted to have sound. And we knew how to do those things. We went in and showed them that we could help them design new machines, really work in partnership with them. And do it even less expensively than what they could in trying to manage software development themselves.

In the case of Apple, Wozniak had done the Integer BASIC. And he was playing around thinking about doing a floating point BASIC. I don't know why he never got around to it, but they knew they needed one, and so they had Jobs and Randy Wiggington come out and talked to us. And I put the cassette extensions in integrated into their ROM. And that became what was called AppleSoft BASIC.

In terms of Commodore PET, they started with us from the very beginning. Because we helped Chuck Pedal, who was at Commodore at that time, really think about the design of the machine. Adding lots of fun characters to the character set, things like smiley faces, and suit symbols. That was the first machine we did that had this wild extended character set. All these machines started out using cassette-based storage, where we could store about 1,200 baud worth of data on these cassette tapes.

So, this was really a generation of machines. These were the popular 8-bit machines. There was another type of 8-bit machine that was a little higher end, that were the CP/M-80 machines. All which ran this operating system from Digital Research. We wrote our languages to run on top of that. That was Gary Kildall's company.

That really defines the 8-bit era of computing. I mean there were some refinements. Like we can see here that Radio Shack built a portable version of their computer. And this is the Osborne computer, that's actually a CP/M-based computer. This is an attempt at portability in this 8-bit era.

We're actually running, on this machine here, one of our first applications. We were broadening out by now to a lot more than just BASIC. We had a portfolio of languages. But we also had a spreadsheet and we were about to come out with a word processor.

So, this is the screen appearance of this spreadsheet called Multiplan. It is kind of an obscure footnote in history, because VisiCalc that came before it, the real creator of that category did incredibly well in the Apple II. We designed our spreadsheet to work on 8-bit machines and the next generation. But a 1-2-3, which was only designed for the 16-bit machines got ahead in terms of features. Multiplan, although contributed a lot of good ideas and, actually, was essentially passed by with the work that Lotus did on 1-2-3.

DKA: Bill, you skipped quickly through a notion of change and strategy in your corporation. First you started with languages and then you started branching out. Can you talk a little bit about the thinking and talking that you and Paul and the others did about what strategy Microsoft as a company should have?

BG: We always knew that we didn't want to have a single product that was a dominant product. We wanted to hire in more software people and have a full product line. In a sense, one of the earliest things we decided to do was to make available on the microprocessor everything that had been available on the mini computer. And that is why we did the languages, COBOL, Assembler, all the normal tools that you would expect to be able to do with native software development. But, it was clear that as an individual machine there were some things that you would use that you didn't need on a mini computer or mainframe.

And already there were other software companies, although, we had been the very first micro computer software company. A company called WordStar had emerged. One of the marketing people from IMSAI, after IMSAI wasn't doing so well, went out and focused that company around the WordStar word processor. Actually, the company was originally called MicroPro. And Seymour Rubinstein ran that. There was Michael Shryer with Electric Pencil. And we had come out with games very early on.

Here is the so called, Adventure, game running here on this TRS-80. This is an early box that we used in our packaging. We called it, that group, Microsoft Consumer Products, at the time.

One of the products that really got us kicked-off in doing new things, was we made a card that plugged into the Apple II with the little Z-80, a microprocessor that succeeded the 8080 on it. So, it let you run all of the business applications that were coming out in this CP/M-80 world, on this machine which was very high-volume. That was a big success and let us grow our retail group, and even go out and license some products like Flight Simulator and many others that defined a whole new avenue for us.

We never saw ourselves as limited, as in terms of what kind of software we wanted to do. As long as it was software where development talent was the key to doing it well. And that it could be sold in fairly high volumes, which in those days weren't nearly what they are today. We decided that might be a good product for Microsoft.

DKA: Do you feel like you saw the movement towards the business market out of the hottest market soon enough?

BG: Well, the key to the move to the business market was getting a machine with more capability, and getting a standard, although you could exchange some basic programs between these 8-bit machines. If you went and used any of the advanced features then you were just tied to that one machine. So, it wasn't all that economical to write lots of software for these machines. And the 8-bit machines maxed out at 64K of memory. And even though I wrote BASIC in 4K, 64K seemed like a lot, but as we started taking on more and more challenges, it just wasn't going to be enough.

And since we were tracking what Intel was doing very closely, in building the next generation chip, their 8086 family, including the very low-cost part called the 8088, we knew that 16-bit computing was on its way. We saw that it could be a good business machine and we decided to focus a lot of early work on that Intel chip. In fact, it was that decision that forced us to do the SoftCard, because we had so many products for the Intel chip, and the question was, "Should we spread those products over to other 8-bit chips, like the 6502 that runs in here? Or, should we immediately move up and do 16-bit software?" And I said, "No. We are going to do 16-bit software." Everybody was a little bit disappointed because it meant that we wouldn't be able to sell onto these machines. That is when Paul invented the idea of the SoftCard, so that we could actually take our Intel software and run it on this machine, and, at the same time go ahead and devote our resources to being way ahead of everybody else in developing software for the 8086.

DKA: You mentioned that you were tracking the developments throughout the hardware side of the industry. How easy was it to get information on what was coming available? Were people open in those days? Or were they pretty secretive?

BG: Well, there was no secrecy at all, at least as far as I could detect. Everybody in the industry would be at shows like the West Coast Computer Fair. And there was so much to do -- we were overlapping each other some. There was some good rivalry, but not in a sense that people were keeping lots of secrets about what they were up to. It was a very exciting business at that time.

Intel was a customer for our BASIC. They came out and asked us to do some custom work. I remember telling them that I could do it in two weeks. And they said, "Don't say that, don't say that -- say four months -- say something reasonable." And it turned out that it took four weeks to do, because configuring their system was so hard. But we'd gotten to know Intel. And we were talking about where the chips were going. It was a very small group that sort of shared this secret. In fact, Ted Nelson gave a speech at one of the West Coast Computer Faires about how we were going to overthrow all the big computer companies, and we really knew that it was "power to the individual". There was a little bit of a minority feel, that we had to advance this cause. And, eventually everybody would realize that we were right about what was going on with these machines.

DKA: Was that shared by the people at Microsoft?

BG: Oh, absolutely. Microsoft at this time, I think our average age was 22 or 23 years old. People enjoyed the programming. Some of the people didn't ever go off to see customers. They just stayed and did the work. But, it was a very small group. And when I came back from a trip, I'd always talk about what I'd been up to and what we were seeing out there in the marketplace.

DKA: Talk a little bit about how your company grew through this period as you started moving around into the business applications, different machines, you added employees. How did that work?

BG: We were in Albuquerque until the end of 1978. And that takes us up to where we really completed all the languages we wanted to do in most of our 8-bit software work. So, as we were moving up to Seattle in 1979, we had some custom extensions to do for some Japanese machines of that time, like a NEC machine and an OKI machine. But, a lot of our resource was already focused on 8086 development software. We just saw that as the coming thing. But moving to Seattle let us expand our personnel quite a bit. That is when the Multiplan, the spreadsheet development started. That is when Microsoft Word started. In particular, we hired a number of key people like Charles Simonyi, who had been a founder of the Xerox Palo Alto research labs, and had shown us the Alto computer and talked to us about graphical interface.

We shared a common view that low-cost microprocessors would be doing that, and got him on board to help us write applications that would eventually become very graphical applications.

So, when I moved from Albuquerque in the very, very beginning of 1979, there were the sixteen people. Then, about a year after that, I hired a friend of mine from college, Steve Ballmer, who is very good at hiring people. He could see that we had more projects that we wanted to do than we could. He was able to almost double the size of the company and people every year for the next five years. So, it really started to change in character where I had written the high percentage of the code myself until we got to Seattle, and reviewed everything that people were doing to the point where we were setting up a lot of autonomous teams, and having to do a lot more in terms of what was our methodology, and how did we interview people. Just to stay up with all the projects we were going after.

DKA: Did you continue to look for the same sort of people?

BG: Well, certainly in development we did. I mean, for the first time we hired a salesperson. We hired a finance person. So, we had some new categories of skill that were things that I had essentially done, but that were more demanding now. We hired in Vern Rayburn to run this consumer products effort, to think about packaging, pricing, promotion, and how to get those things out there. He hired quite a few people to help him work in that area. The Development Group, which has always been the core of Microsoft, there it was the same kind of person coming on board, but the company had more aspects. In fact, we started our activities in Japan and Europe. So, we were getting people in many different places. Even though Development we kept all here in Seattle.

DKA: Was everyone in the company happy about the move to Seattle? Or was that a matter of concern for many people?

BG: Well, it took a while for me to sell the idea to everybody. But, in fact, everyone, except my secretary, decided to make the move. Albuquerque has its advantages; warm weather, a nice place. So, it took quite a bit of selling. But, everybody was very involved in what they were doing, and there was the excitement behind where we were going. So, I was able to get literally everyone but my secretary, Miriam Lubow, to come up with us. And that became the core group. But Seattle was a lot better for hiring, particularly once Steve got in and really increased the rate a great deal.

Our hiring was always focused on people right out of school. We had a few key hires like Charles Simonyi who came in with experience. But most of our developers, we decided that we wanted them to come with clear minds, not polluted by some other approach, to learn the way that we liked to develop software, and to put the kind of energy into it that we thought was key.

DKA: What would be the two or three things that would characterize the Microsoft way of computer software?

BG: The key for us, number one, has always been hiring very smart people. There is no way of getting around, that in terms of I.Q., you've got to be very elitist in picking the people who deserve to write software. Ninety-five percent of the people shouldn't write complex software. And using small teams helps a lot.

You've got to give great tools to those small teams. So, pick good people, use small teams, give them excellent tools; vast compilation, debugging, lots of machines, profiling technology, so that they are very productive in terms of what they are doing. Make it very clear what they can do to change the spec. Make them feel like they are very much in control of it.

Have lots of people read the code so that you don't end up with one person who is kind of hiding the fact that they can't solve a problem. Design speed in from the beginning. A lot of things that have helped us, even as the project teams have become larger, and the company has become a lot larger than it was. It is not some methodology where there is a lot of funny documentation. Source code itself is where you should put all your thoughts, not in any other thing. So, our source codes, all though there are a few exceptions, tend to be very well commented in a very structured way.

DKA: We talked a lot about some of your earlier successes. What were some of the early things that didn't go well? And what did you learn from them?

BG: There were a lot of mis-steps in the early days, but because we got in early we got to make more mistakes than other people. I had customers who went bankrupt and didn't pay us, customers who we spent a lot of time with who never built microcomputer-based machines.

I worked for a long time on an APL Interpreter and I almost got that done, but then it looked like it wasn't going to have much of a market and I was too busy doing other things. We actually never shipped an APL that we had talked to people, that we were working on.

There was another interpretive language called FOCAL that we'd written a version of for the 8080 and 6502. Having two interpretive languages like that was not a good approach. That was a dead-end project. Everything else, COBOL, FORTRAN, the way we selected the various chips. A big project we did for Texas Instruments where we wrote a product for their machines which eventually, they failed in the personal computer market, but they shipped a lot of machines in the meantime.

Multiplan, targeting the 8-bit machines instead of just relying on the next generation to come, the IBM PC generation, that was a huge error. When we talk about, "Are we aiming too low, in terms of system requirements, we often think, is this another case like Multiplan?" Because it was a great product, but it was the basic strategy that was wrong. And, in fact, to some degree that allowed me to make one of the best decisions I ever did, which was later, when we had to compete with 1-2-3. There was a question of whether to do it in the character-mode environment, or whether to move up to the next generation, which was graphical. And we said, "Okay, we'll let them dominate the DOS-character world. We are going after Mac and Windows. We are going to be a generation ahead." And that worked out very well. Multiplan was certainly an experience that was helpful there.

In Retail Marketing, we made a number of mistakes that were important for us to learn from. We had in a few countries, agents. And you really don't want to use agents. You want to have your own people. If you are going to be a serious company, take a long-term approach. You should hire people in all the countries you are going to be in and make sure they are there cementing long-term relationships -- not just generating short-term commissions. I think we learned that one pretty quickly. We did hire in some very sharp business people, and got them to share their experience so it wasn't just us technical guys and the other people. We were very young. I mean, Steve and I were kind of driving the business and Paul and I were driving the technology. We were optimistic in thinking we could get things done sometimes faster than what we did. The project of the moment always seemed very exciting. And some of them never generated much in the way of royalties. But all correctable stuff as long as we sort of wake up and see what the results were.

DKA: You mentioned your relationships with Steve and with Paul. How did that set of relationships shape and change as the company grew and spread?

BG: Well Paul, of course, was my friend from the early Lakeside days. And we are very close friends today and I'm sure we will always will be[Bill laughs]. He is very idea-oriented. He and I would brainstorm about things. So even though I was running the business, it was a partnership. His role was very, very critical to so many of the transitions that we made. There was always some strain because I was pushing people to work hard, including Paul. That wasn't really a big problem.

When Steve came in, I was spending more time with him because the business side was important; managing and organizing, and (deciding) what we were going to do about international. So, it was great that Steve was smart enough and personal enough, that even though he didn't have a technical background, the programmers accepted him. That was very rare. We didn't really believe non-programmers should manage programmers. And we didn't do that until I think about 1983 before Steve actually directly managed developers. But the developers accepted him early on because he was smart, he would sit and listen to them, understand the things that they really liked to do. And so that fit in. I got a lot of benefit out of Steve going around and always knowing what people were thinking about.

And there were other people. Kazuhiko Nishi, who is a very close friend of mine from Japan, really taught me about the Japanese market. He got us doing the very first Japanese personal computer that any NEC PC-8000 and many of those other projects. He is a visionary, very energetic, almost overly optimistic about where things can go. He started a lot of the early computer magazines in Japan, and worked with us for a long time.

Because it was a fast moving business, although we worked very long hours, we'd go out to movies together. Everybody knew each other awfully well. Up until 1981, Paul and I had owned the whole company, except that we'd shared a little bit of it with Steve. There wasn't much pressure to do anything differently than that, but then around 1981 we decided to share some of the ownership in the form of options. And actually brought some of the really strong contributors into that program.

DKA: Bill, you're famous for a vision that you had about personal computers. Can you tell us about the vision?

BG: The vision is really that in the information age that the microprocessor-based machine, the PC, along with great software, can become sort of the ultimate tool dealing with not just text, but numbers and pictures, and eventually, even difficult things like motion video. And that is something that when Paul and I would go around speaking about computers, we would always say that there were no limits. We used to call it the "MiPs to the Moon" speech. That performance would be unbounded and that all of these incredible things would happen. We were never too specific about exactly when various things would happen. And, of course, when we went back to our business we had to decide what our priorities were. But, the frontiers were sort of wide open. It was that sense of excitement that we really wanted to spark in everybody else wherever we went.

DKA: Well, you were both a visionary and a pragmatic business person. How did the competition rank in these early days? And how did you manage to continue to make the progress against your competition?

BG: Well, there certainly were a lot of other software companies. Within two or three years of our being started, there were dozens of companies. Some of them tried to do better BASICs. And we made darn sure they never came near to what we had done. There were competitors in other languages. There was Digital Research, VisiCorp with VisiCalc, MicroPro with WordStar, Ashton-Tate with dBase. And each of those companies made some huge contributions, very innovative things. Lots of game software companies and that was an area we chose not to get into. Lots of people doing applications packages like payroll. We'd toyed with getting into that, but decided that it just didn't leverage what we were good at well enough--and that we should probably let other people do those things.

There were many fine companies. I'm probably skipping over hundreds of huge contributors in the software arena. They didn't take quite the same long-term approach that we did. Doing multiple products, really being able to hire people and train them to come in and do great work, taking a worldwide approach, thinking of how the various products could work together.

So, we were more comprehensive. We weren't the largest. There was a time that MicroPro with WordStar was bigger. There was a time when Visicorp was bigger. There was a time when Lotus, with the early years of 1-2-3s incredible success, was bigger than we were. But we were always the most technical. Whenever anybody else in the software industry wanted to know where we thought things were going, they'd come and talk to us. Because our vision, we shared; we didn't view that as some competitive edge. We just wanted to talk about it and get other people to share the same ideas so that they would help make it all come true.

DKA: One of the most interesting machines that came out of this area was the TRS-80 Model 100. Do you want to say a few words about Microsoft's role with that machine?

BG: Yes. This is in a sense my favorite machine, I mean by today's standards it is kind of a pathetic machine. But what happened was Kazuhiko Nishi, my friend from Japan, came over and said that we could have an 8-line LCD with 40 characters. And up to then all we had was four lines by 20 characters. I didn't think using 4 by 20 you could do much that was interesting. But, when he said we could go 8 by 40, then I got to be pretty fascinated with the idea of a portable machine. It wasn't just taking your desktop machine and trying to shrink it down, because battery life would be a problem, and ease of use would be a problem. But just taking the things you want as you move around and making it pretty inexpensive.

So, this machine came out for \$500. Jey Suzuki, from Japan, and I, wrote the ROM in this machine. It is a 32K ROM. Part of my nostalgia about this machine is this was the last machine where I wrote a very high percentage of the code in the product. I did all the design and debugging along with Jey. And it is a cool user interface, because although most of the code is a BASIC Interpreter, we did this little file system where you never had to think about saving anything. You just had this menu where you pointed to things. It was a great little editor and scheduler. We crammed it all into a 32K ROM. And really designed it in an easy to use way around these special keys up here. This machine was incredibly popular with journalists. Even though it came out over 11 years ago now, it was out by 1982. You still see some journalists using this, although the technology has gone way beyond it.

We had some great things here like we had a way that you could add a bar code reader to this. We thought maybe people would distribute software on bar codes. In fact, Byte Magazine got into that for a while. We had a lot of ways you could extend this by putting a new ROM in the bottom. And it was sold not only in the U.S. by Radio Shack, but NEC sold it in Japan, and Olivetti sold it in Europe. And the company who made it, Kyrocera, became a good partner of ours for lots of future projects.

DKA: You may actually want to turn it on so that we can show it.

BG: Let's make sure that this machine is still running. My God, it's a machine that works! I don't know how LCDs work in a camera. What you had here is just your files. And you would just move the cursor to the one that you wanted and hit the Enter key. And then you'd be back editing that file. So, if we go into text, you can type in the name of the program and it would know that's what you wanted. It is a nice screen editor. You can just move the cursor around. The only real problem with this product is that the keyboard was noisy enough that if you sat in a meeting with it, it was still considered anti-social because you'd just be tapping away during the meeting. So actually we did a version, just a slight modification, soon after it came out that had a very silent keyboard so that people could sit in meetings and use it. It is really a nice machine. A great, great way that we use these function keys.

DKA: That kind of wraps up the number of topics we wanted to treat today. Are there any parting comments that you have about this era that you'd like to record before we turn to the PC tomorrow?

BG: No, I think the PC era really can be divided pretty nicely into the pre-IBM PC era, and the post-PC era, because as soon as the PC comes out, as nice as all of these machines we've talked about are, they faded from view very quickly. And with the really sole exception of Apple most of the companies involved faded from view. Most of the software companies who had been focused on these machines, with, of course, the exception of Microsoft.

So, there was kind of a hobbyist focus here, even though we got diskettes, there were some of examples of businesses using these things. They weren't reliable enough or the capacity wasn't enough, or just people hadn't adjusted enough to really think of these as a key business tool. The Apple II is emerging a little bit in that way. But before business would really buy into this thing, it took that next era, which was a whole new level of excitement.

DKA: Bill, do you just want to tell the story of this paper tape [referring to the Altair BASIC tape]?

BG: Sure. Besides entering programs into the front panel through the switches, there is always a question of how to send software around. And because Teletypes were reasonably inexpensive, being used in direct communication networks even before PCs came along, the 8-channel paper tape became a key medium of exchange. Now, when a Teletype would read this in, it would only read at ten characters per second. So, when you are loading 8K BASIC it would take about ten minutes for it to read into the computer, and you would hope that it would read in properly.

This tape here is actually the very first BASIC that we did. That BASIC, in total, was about 4K of code. I had to squeeze a little later so that you could run with code and the program in 4K. But at first it was about 4K. So, you see it is about half the size of 8K BASIC. We sold BASIC in this form for most of the first year and a half. This was the dominant form. But then things moved over to cassette and that was the popular form for another year and a half. Then we started to get floppy disk in. We tried a lot of things like reel-to-reel tape, and various digital tape cassettes that people came along with. But it was really the floppy that defined the next level beyond the cassette and paper tape.

DKA: You were telling us yesterday that you actually stayed up just about all night working on this tape. Do you remember what it was like to actually punch that tape before you sent it to Albuquerque?

BG: Well, the PDP-10 that we did the development on actually had a high-speed tape punch, so it could punch the tape in about three minutes. So, the issue wasn't so much the tape as it was making sure everything that was on there was right. I actually used fan fold paper tape, it is the same, but they fold it every six inches or so. And I punched three different copies out on the PDP, which is the peripheral name of a paper tape punch on the DEC 10. I made sure that Paul had all of those in case there was any kind of problem. But paper tape was actually quite reliable. It was just slow to work with and easy to mishandle. I mean you have a huge paper tape. You're always kind of dropping it. And it would get twisted up, things like that.

DKA: So, really, what you have in your hands is the beginning of your company.

BG: Yes, this BASIC was the first real piece of software ever written for a PC. And it became for the first generation of PCs, the thing that unlocked the power that was there, because, although some people did machine language programming, 90% of what was done was done in BASIC. And 90% of that was Microsoft BASIC. But, descendants of this tape got onto all those early machines.

DKA: Bill, when we left off yesterday, we were talking about the PC, the IBM PC, as beginning a new era in the history of the personal computer. Why is that so?

BG: Microsoft started focusing on developing 16-bit software as early as 1979. We decided that what Intel was doing with their 8086 was really the way to go. Actually, Motorola had played around with the 68000 design and decided not to go forward aggressively with that. Instead, they did the best 8-bit processor ever done, the 6809. And we did a BASIC for that. I liked that chip. But in terms of really going after a larger address base, this 8086 family was it. And so even before IBM came to us, we worked with a number of companies, including a little local company called Seattle Computer Products or SCP, to show that we were moving software over and writing new software for the 8086. It was higher speed and it broke the 64K barrier of the previous machines.

We also thought that in moving up to this new level, instead of having each manufacturer go and do different features so the software's special, we'd have a chance to set a binary standard so that the same system software and all the same programs would run on one machine. So, we were looking for a chance to do that. The first companies to do 8086-based machines, big companies, were Mitsubishi in Japan and Victor. Both did very nice 8086-based machines with good graphics. But the seminal event certainly was the group at IBM that pulled the machine together. It is kind of a long story that has been talked about where they were chartered to do a home PC, but even more importantly they were chartered to do something quickly.

IBM was frustrated that they were taking five years between product concept and shipment. So, this group had sort of bid to get the charter to do the PC based on saying that they could do it in less than two years. In fact, they had said they could probably do it in a year. But the trick was to go to outside suppliers for the key components. So, they went to Intel to get the chip. And they came to Microsoft to talk about, not only software, but the overall system design. They'd seen that we were in all the personal computers up to that point, and read about us before they came up. Told us that we had to sign quite a non-disclosure. These were just product planning people. But we talked about our enthusiasm for a 16-bit PC. That they could go and told them they ought to talk to Intel and it could be even better than the Apple II or the CP/M-80 machines which were sort of the higher-end popular designs of that time. So, the project started in late 1980.

The very first issue of PC Magazine came out with the story about the work between IBM and Microsoft. Dave Bunnell had been at MITS on the Altair and I had gotten to be good friends with him. He actually started up a PC magazine after he left MITS, *Personal Computing*, and I had done a column for him. So, when we were working on the development of the PC, I encouraged him to go out and do a magazine around it, and he worked, actually, with Tony Gold who'd had LifeBoat Software, and started up this magazine. So, he came out and interviewed me and I talked about the whole project.

It really started when they came out in the summer of 1980. It is pretty amazing. They came out in the summer of 1980. We talked, agreed to work together by September. Started engineering work. Actually signed the contract in December, froze the ROM, which was the BASIC built-in to this, there's a ROM BASIC in here [gestures to the IBM PC] in March and the product shipped into the market in either August or September of 1981.

They had actually a fairly small team. So, in a sense we were working with IBM, but in a sense we working with a small team. The small team included people who had used PCs before. The Chief Engineer of this, and the person who deserves, at the hardware level, an immense amount of credit is a guy named Lew Eggebrecht. He later got into some disagreements with IBM and left and sort of never got his due, in my view, for his role.

The team, of course, was headed after Bill Lowe got the charter. It was handed over to Don Estridge. And so it was Don who I worked with a lot, and the people who were working for Don. It was a very close partnership at that level because they were a little naive and very open-minded. They told us for this keyboard, they said Lexington had to use a certain layout. We wanted these function keys to be up here, that the Lexington layout forced them to be over here. We ended up with this funny big key over here because of Lexington. But here we'd put in our favorite fully-extended character set. Came up with some new ideas there. Put some special things in here that we thought we'd be able to use. In fact, we put in the Wang word processing character set because we weren't sure whether to do a Wang-clone word processor, or start with our own approach at that time.

There were many issues about this machine. Should it have graphics or not? First design did not. We convinced Eggebrecht that he could do a graphics card. The funny thing was, they wanted to have these good looking characters. So, they did a video card, this character mode card that didn't support graphics, but they did go ahead and do the color graphics adapter, CGA card. The graphics weren't very good, 640 by 200. The characters were only 8 by 8. So they don't look as good as this character display does. But that was fundamental. If they hadn't done that, there are so many of the great things about the PC that wouldn't have been possible.

DKA: What was it like to work with IBM? They were a very different company from the companies you had worked with before.

BG: Well, IBM, as a big company, was very different. It meant the non-disclosure, the contract, the security measures, the way we related to their various overseas groups. That brought in the Big IBM. But the individual people, whether it was Sandy Meade, Mel Hallerman, Dave Bradley, they were all reasonably similar to us. A little bit older, but also pretty hands-on, pretty energetic. And a small group did this. We actually had more people assigned to the project than IBM did. And we were only a company of 30 people. We threw ourselves into this. Although we had first just started talking about the BASIC, but then they said they wanted other languages.

We knew that we had helped design a machine that was more than a home computer. So, then we were kind of saying, "Well, if you want the other languages, you must know too, this is not just a home computer. Let's not joke around." You don't put COBOL and FORTRAN and all that on this machine. And so that group and Microsoft sort of conspired to make it an all-purpose machine, not just a home machine. The base machine came without a floppy disk. No disk at all. And 48K of RAM and the ROM BASIC. They didn't ever sell many machines, but actually in terms of designing this thing, that was very tricky to get everything to work in that much memory and have the ROM so it could work without the operating system there, but also work when the operating system came in and yet be patchable to the ROM if anything was ever wrong. Very tricky issues.

Some interesting things include: the way we had decided that the schedule was so tight that we would refer IBM to go down to talk to Digital Research about CP/M. Well, they didn't want to sign the non-disclosure and they really didn't jump on it. So, the project was at risk. It kept having to go through IBM reviews. And if they didn't have all of the software signed up they clearly weren't going to make the schedule. So, we said to IBM that we could do that. And as an increment on top of what we had committed to do, it was about ten percent extra work. So, we went out and bought from Seattle Computer Products the work that they had done on an operating system they called the SCP-DOS or 8-DOS at the time. But, more importantly, we got Tim Paterson to come across and work for us. He was the primary creator of MS-DOS. Bob O'Rear was one of our people who worked very closely with him and put that together.

I was very involved in the creation of the BASIC. Actually, Paul Allen got involved in that, and a number of other people. But, when it came right down to it, everybody pitched in, Neil Konzen, myself, Paul. There was a very tricky design. For example, proving to them that we could do some great stuff in the graphics area so that it was worth doing this CGA card. Proving to them that we could use the palette.

We were able to take a lot of innovative work we'd been doing on machines in Japan, OKI F-800, Hitachi BASIC Master Level III, NEC PC8800 and take that and put it all together here because this was a 16-bit machine. And so it came out very, very well. They didn't invite us to the introduction. That was kind of a unique IBM thing. But we had a great relationship. They were down in Boca Raton. So, we were taking lots of all night flights down there. They kept the machine locked away in this hot room, which was a real problem in terms of working with it. Whenever there seem to be security leaks, they'd come and ask us. Actually, Kazuhiko Nishi did some of the design. He made sure they put speakers in. We had a primitive sound capability that was Kazuhiko's idea. So, they were worried -- were the leaks in Japan due to him? Very intense project.

DKA: We should hone in a little bit, Bill, on the MS-DOS development. You wanted to show us a little bit of that. Your involvement with MS-DOS shaped a lot of what would happen later. So, let's just talk a little bit about -- you've highlighted the issues going into it. But just taking some of the fundamental ideas were that reflected MS-DOS that are still part of personal computing today.

BG: MS-DOS, although in its first version was fairly limited. It had to run, as I said, in that 48K machine. It went on to become the foundation for this binary standard where so many applications would come along. It was not the only operating system brought out with the machine. They actually brought out CP/M-86 which was when Digital Research woke up to 16-bit, what they went and did. And another thing called the UCSDP system. But the one that shipped first with the machine and the one that was low priced, because we didn't insist on significant payments, because we wanted it to get out there, was called the IBM Disk Operating System.

Now, the funny thing is IBM didn't like these acronyms. So, in fact, even the term PC or words like PC DOS, they at first really didn't like people using those. But it became so commonplace that the operating system was sometimes called PC DOS, sometimes called MS-DOS. And, of course, the machine became the PC and all the magazines about it were the PC magazines.

We worked with IBM to have some software available from the very beginning. We are seeing here some of these BASIC sample programs. Neil Konzen and I spent a whole weekend just goofing around with this machine once we had BASIC running and wrote this thing, DONKEY.BASE, PIECHART.BASE, these are still shipped today. They are kind of obsolete, but they are a code that we threw together that weekend. The key to the competition in operating systems was getting lots of applications. And Microsoft got very serious about working with manufacturers and software developers. MS-DOS started out a little bit behind because it was easier to take 8-bit programs from CP/M-80 to CP/M-86. So, we came up with tools to help with that. And it was a real competition, I mean although it is an obscure footnote in history now, from 1981 to 1983, people weren't sure which system would win.

Even until as late as 1986 some companies like Digital Equipment with their machine, the Rainbow, were very focused on CP/M-86 and not on Microsoft MS-DOS. And then by 1986, Digital Research kind of stopped really trying to compete and MS-DOS really pushed ahead.

DKA: Let's talk about one of your principle applications that came fairly early on that was the development of a word processor, Microsoft Word. What characterized Microsoft Word?

BG: Microsoft Word, although it was the second application that we came out with after Multiplan, was really a milestone because with this one we decided we'd really do something forward, very forward looking. We'd hired Charles Simonyi from Xerox PARC. We knew that graphics interface was where it's at. We knew that laser printers were going to be very big. So we designed something whose underlying structure was ready for the graphical world. In fact, we made it so that it could show italic and bold, it could work with the mouse, and it was laying things out internally to immense precision. So, even though the screen didn't have that precision it would print with very high quality. We were making it ready for the laser printers that were going to come. The user interface here was identical to the Multiplan user interface.

There was a family of products that we called the Multi-Tool Family of products. There was Multi-File and Multi-Chart. Those were Microsoft's character-mode applications. We saw that selling these applications would actually be as big a business for us as doing the operating system itself. Our color at the time was green. So this is the documentation we came out with for Word. And although Word was very much praised for its technical richness and anybody who got to use features that had these so called "style sheets" and the ability to use this mouse pointing device wasn't necessarily as easy to use because of the manual and other products that were out there. So, although MicroPro with WordStar really started to fall apart, the vacuum was filled by a number of products; Word to some degree, Aston-Tate had their MultiMate product that they had bought. There were a number of products.

But WordPerfect over time really took the role that WordStar had had. So, 1-2-3 and WordPerfect became strong products, partly with the introduction of the PC. 1-2-3 came along a little more than a year after the PC had shipped, in January 1983. And that really helped to push things along. Because this machine was from IBM that helped. But people who think historically think, well, IBM was preordained to dominate the market. There is just no way that's true. IBM succeeded because they had a good team. They were the first to be serious about a 16-bit machine. They worked with us. They worked with Intel, with Sears, and Computerland.

And, perhaps most importantly, this machine became the model. This was the machine that people decided to clone in order to build this base of total compatibility. And because people had done that to IBM mainframes it was perhaps most natural to go and do it with this particular machine. And so by making the bus and the software and everything in here licensable by other people, they setup a thing that the whole industry could rally around as the next generation of 16-bit computing.

DKA: Now Microsoft is primarily a software company, but you actually got into some important hardware development with the Mouse. Do you want to say a few words about that?

BG: Microsoft was playing a much broader role[laughs] than just doing software for this machine. I mean whether it is the keyboard, the character set, the graphics adapter, or even the memory layouts. I laid out memory so the bottom 640K was general purpose RAM and the upper 384 I reserved for video and ROM, and things like that. That is why they talk about the 640K limit. It is actually a limit, not of the software, in any way, shape, or form, it is the limit of the microprocessor. That thing generates addresses, 20-bits addresses, that only can address a megabyte of memory. And, therefore, all the applications are tied to that limit. It was ten times what we had before. But to my surprise, we ran out of that address base for applications within -- oh five or six years people were complaining.

Another thing that Microsoft did, in terms of getting these new machines out there and really showing off what new uses they could be put to, was we came out with our own mouse product. The mouse was invented by Doug Englebart back at Stanford Research Institute. Xerox used it in the Alto, the research machine that PARC built, has a three-button device. The Star had a two-button mouse. And then went Apple went and did a mouse, they did a single-button. We believed the two-button concept was the right approach. So, we went to a Japanese company, Alps, got them to do some design work, paid the patent fees to SRI and Xerox for this, and came out with this as a low-cost add-on. So, even on a character-mode display, being able to move the cursor around in a natural way, we thought was a big advantage. We tied it to Word so that we had a bundle with Word and the Mouse. But then people who didn't like the Mouse thought they shouldn't buy Word. So, it was a little bit of a problem.

When we first brought this out we ordered 50,000 and it took over a year to sell the first 50,000. Today we sell many, many hundreds of thousands in a month. But, at first it looked like maybe we had made a mistake. This did go on to be a very profitable thing for us and we continued to evolve the design going to a sleeker and sleeker appearance over time.

DKA: Bill, a number of other people quickly got into the business of making IBM-compatible computers. How did that effect Microsoft?

BG: This phenomena that took place right after the PC came out, people built compatible machines, was very important. It meant that the software industry instead of redeveloping for every machine, every keyboard layout, every screen resolution, or any new documentation, that you could come out with one disk and it would work in the machines of multiple manufacturers. So, it created the conditions for an explosion in the software business. It took time. The early people to build compatible machines, except for Compaq, actually aren't with us today. Companies like Columbia or... -- I don't even remember the names. There were a lot of them that came and went quite rapidly. Even companies that we think of today like AST, was an add-in board maker back in the early days. And only later got into systems.

Radio Shack continued to make their own design systems for quite awhile, but then eventually built compatible systems. But that phenomena was fantastic because it meant the competition to make the fastest machine, the cheapest machine, the most portable machine. People were moving in so many dimensions all at once. Compaq distinguished themselves with this portable machine. By today's standards, of course, that's a joke. I mean if somebody lugged this onto an airplane you'd have to laugh at them -- "What the heck is it?" But it was a significant advance. Because they sometimes say that the handle was their most innovative feature. So, the PC industry really heated up. The volumes took off. The prices started to come down. These early machines were \$3,000 - \$4,000 dollar machines, and that's without a hard disk, without so much of the capabilities that we expect in a PC today.

DKA: How was your relationship with Apple different from what it had been with IBM and other makers of compatibles?

BG: The relationship between Microsoft and Apple can be divided into a couple of phases. In the early days when I did software for the Apple II, they were quite small and I worked with their Development Group. We wrote a lot of applications that ran on the Apple II. The SoftCard helped the Apple II a great deal. Decathlon Software. Multiplan software. It was a good relationship. But mostly, we were selling to their customers.

The really unique thing we got into was when Steve Jobs came up and talked about what he was doing with the Macintosh. He solicited us to write a family of applications for that machine. And because of our background in looking at what Xerox had done with graphics interface, we were very excited about this design. That was different than the Lisa Group. The Lisa Group had done their own applications. And they'd gotten very big, and were bundled in with the machine. So there was no opportunity there. And the Lisa didn't do very well.

But, what Steve was doing with the Macintosh, pricewise in keeping things fairly lean, looked like a great approach. So, instead of really attacking 1-2-3 just at the DOS level, we decided we would focus on the graphical version and do the work on Macintosh and Windows and sort of be a generation ahead if we were right about graphical interface. So Microsoft worked very closely with the early Macintosh team.

We were their testing group. They had no testers. We helped shaped the features of the machine. Initially, they had wanted to use the floppy disk of the Apple II, 143K disk. That didn't look too good. Then they wanted to use a 5-1/4" floppy where there were heads on both sides, so called "Twiggy Drive." And that didn't work too well. Finally, at the last minute they, switched to this 3-1/2" drive. And that was a fairly innovative thing. HP had been doing that a little earlier, but this was a milestone. The project kept slipping. When Steve first came to us it was supposed to ship in early '82 at \$995 dollars. Well, it ended up shipping in January of 1984 at \$2,495 dollars. Partly Scully had come in and convinced Steve that the demand would be high and that they had to fund a lot of things. So the price edged up a little bit. Also, Steve tried to convince everybody that this could be a 64K machine. And his engineers were saying, "No way, that's too tight." And we said, "No, we can't do any applications." He went up to 128K.

The model we are looking at here called the "Plus" actually came out about a year after the Macintosh shipped. And they brought it from 128K up to 512K. That was because the... only Microsoft and Apple succeeded to get applications in the 128K machine. It was very, very difficult. The day the machine shipped we had our BASIC, our Multiplan, and they had Write and Paint. Those were really only the important applications for the machine for quite some time. A year later they beefed up to 512K and got the laser printer, that's when we started coming out with Word, File, Chart, and then later with the Excel product in 1985, which was a very important milestone for this machine. It meant that it wasn't just for wimps, or sometimes they said, "for the rest of us", but it was for anybody who wanted the most powerful machine that had graphics interface.

The Mac, although today, is so clearly a grand success, but had a tough time in the early days. The first year it sold to enthusiasts. The second year it actually sold a little bit less. Steve Jobs left Apple under controversial circumstances. It really took until 1985 or so, with things like Excel, before it became solidly entrenched.

The Macintosh was a very, very important milestone. Not only because it established Apple as a key player in helping to find new ideas in the personal computer, but also because it ushered in graphical interface. It is hard to reconstruct, but people didn't believe in graphical interface. And Apple bet their company on it, and that is why we got so involved in building applications for the Macintosh early on. We thought they were right. And we really bet our success on it as well. And today, all of the machines work that way because it is so much more natural. But this was pushing the limit.

And it is the first time the vision of PARC had gotten into a commercial product and got going in a very positive way. Interestingly, Windows was being developed in parallel with the Macintosh. We actually announced Windows a few months before the Macintosh shipped. We announced in 1983 with a lot of other manufacturers, although, it took some time before we actually shipped Windows, that we were exposing it and evangelizing it to everybody very early on.

The Macintosh, because it was designed around graphical interface had an advantage. And so it would take as much as five or six years before the PC platform had graphics as a standard feature. It wasn't until 1990, with some developments that came later, that graphics really worked. And so this machine got out in front and had a huge impact.

DKA: Bill, what led to your decision to build your own site, your own corporate campus?

BG: Well, I was always thinking that environment that we did product development in should be a fun environment, a lot like a college campus. And this idea of using small teams means you want to give them all the tools, all the computers, an individual office, whatever it takes so that they feel like they can concentrate on their jobs and be very creative. And, in the Northwest, having a lot of trees around, you know, one, two, and three story buildings where offices are very good sized. That made sense to me. And we had been looking ever since we moved up to Seattle for a piece of land that wasn't too far away and yet that would let us grow as a company. And in 1986 we actually got to move into our corporate campus.

This kind of shows you one of the buildings here. Initially, there were four buildings like this clustered around the lake you see here, and each of the main Development groups each got their own building. And that meant that we really had the best of all worlds. People felt that it was a fun environment, but yet we were really close to each other as far as working together. Things like people juggling or riding unicycles around, having barbecues outside, having company meetings where everybody would stand around. This was the original ground breaking. And these are two of the developers, but here we see Jon Shirley who started with the company in 1983. He came from Radio Shack and played a very critical role, because, although I had Steve Ballmer helping me think through business issues and a lot of other people like Kazuhiko Nishi on the product side, or Paul Allen, it was Jon who helped us really grow; what kind of systems did we have to have in place. And Jon had been my good friend when he worked at Radio Shack, so we were excited to have him come on board. Actually, he was the second president of Microsoft. We hired another gentlemen[chuckles] who was with us for a little less than a year where it wasn't a good match. A gentlemen from Tektronix. And then brought in Jon after that.

There was a week in 1986 that was pretty exciting because that was the week we moved into the campus. Here is Jon with his "Move" T-shirt and I guess this is me in my new office with my "Move" T-shirt. But that same week we moved in, we went public as a company, this is our offering prospectus. And that was also the week of Microsoft's first CD-ROM Conference where we were pushing the idea of multimedia back in 1986 that didn't really catch on, you could say, until 1994 so before it was in the mainstream. But this kind of shows you the pace of activity at that time. We felt fine to have all those things happen all at once. And I know that next Monday I flew down to Australia to be part of a big software show down there.

DKA: Let's talk a little about going public and whether that was something that everybody accepted that you needed to do, whether it was a controversial issue, and what difference it made.

BG: Well, Microsoft had started giving out stock options to people as early as 1981. So we were sharing in the success we thought we'd have. As we did that, they had about a five year vesting period. And so as some people were starting to vest on quite a bit of their stock, there was the question of how would they get liquidity. Now you could just let it be traded privately, but then the price would fluctuate a lot because the supply would be so short. And I was quite reluctant to go public because of the overhead. We had been able to track our stock price internally up in a very linear way. And with the market sort of maybe over-anticipating the future, or getting paranoid -- you know the stock would be very volatile. But I was convinced that it made sense. And as long as we were going to do it, it was an opportunity to really expose the company broadly. Talk about our vision where we had done well, where we were taking the industry. And it did become something that was covered very, very broadly by a lot of people. And it was extremely successful. The stock took off after this offering at \$21.00 and it just zoomed up from there for many, many years.

DKA: There was no regret about going public?

BG: Going public is not without its complexities in terms of dealing with analysts and all the reports. It is a little convoluted about when can you keep things secret versus having to go out and talk about those things. So, it is not totally simple. But the benefit of having the stock be very liquid for everyone was very positive. We didn't use any of the money that we raised. We just put that money in the bank and it sat there with all of the money that we had earned, because we were very profitable and had plenty of cash by this time. So our reason for going public was very different than any other company that was going public.

DKA: You have a couple of charts there, Bill, that kind of track your growth that you might want to pick up. I'm curious to know whether it was more satisfying for you to add your first 100, 1,000, or 10,000 employees?? And, how did you feel about the incredible growth in your company in this time period?

BG: Well, the growth in sales was pretty rapid. We were doubling every two years. It just looks like an exponential curve. It is unusual because in here there is a lot of products that are coming and going, and products that are doing incredibly well and yet it looks very well ordered. The profits went up the same way. Actually, the profitability percentage rose a little bit as we moved up here. So it was very tightly managed. The thing that really changes the company isn't the sales growth, though, it is the number of people. This shows you the total here. We are now well over 15,000 employees. And yet, until we moved to Seattle we were about sixteen employees. Then we got up to a little over 30 when Steve came.

Then right after Steve Ballmer comes then you see these periods of 40 to 129, 220 - 476 increasing at a very rapid rate. It is swell because you can have all these other product groups. You can sell around the world and you can do better product support. You can lavish on the developers a great library. You can start a research group that is looking way out into the future. And your recruiting department can be the best at going out to all of those universities. So many excellent things that the energetic, smart people we hired empowered us to do. But it does mean that I got to the point where I couldn't look at all of the code, which I had done in the early years.

At 100 people, I knew everybody. I even knew their license plates when they came and went. I knew really what everyone was up to. By the time it got to 1,000 that was no longer the case. I was hiring the managers and knew all of the managers, but there was a level of indirection. And, certainly, as you go up over 10,000 then there are several levels indirect. There are some managers you don't know. There are some products that you certainly know how they fit in if you are setting the overall strategy. Now electronic mail has been a huge phenomena for us. And it keeps a little bit of a closer feel even if somebody's office is in another building, you're always sending them messages. Or, even if they're off in another country it makes that easy. So, we are not as big as our size suggests. But, it certainly has a bit of a different feel than the small company.

DKA: At the same time, has your culture changed? Or has it been pretty stable?

BG: Well, the culture, of course, varies by group. The sales force has great sales people and have the same enthusiasm for software, but their day-to-day activity and measurement is quite different. In the product groups it is pretty similar today. The developers like writing great code. They like collaborating with the team, they like getting things out every eighteen months. They like watching the competition.

If somebody is very smart and contributing a lot, then it is fun. If they don't match that kind of level of energy, then it is really not the right place for them. It is an exciting thing. It is still a little bit different. People can't come and talk to me everyday[laughs]. And so they have to look to their Business Unit Manager, which is how we have it setup. Certainly, we are trying to preserve all of that culture, and get the advantages of being a large company with a broad product line, with stability, worldwide presence, great support, and yet have the advantages that a small software company has.

DKA: Bill, with the AT came a new chip that brought a lot of traditional speed and capability. How did that affect your business at Microsoft?

BG: After the PC came out in 1981, the next step that IBM took was the XT where they put a hard disk in. So, we did a version of MS-DOS with a hierarchical tree oriented file system. The next step after that was the AT 286-based machine. The 286 was quite a bit faster than the 8088 that had come before. Almost four times as fast as the original 8088 machine. The problem was that if you wanted the 286 to address more memory, it had to run in a mode that was completely incompatible with the original chip. And none of the software would automatically move over to that. And this was because the 286 chip was designed before the PC got popular. So, you could either use the chip and emulate the old machine, and just have the speed. Or, you could run it in the incompatible mode and get extra memory.

Now, there are a lot of crazy ideas about how to get around this limitation. IBM and Intel fooled around with one that was kind of an emulation trick that involved using the debug features of the chip. It didn't work for them, so they got me to help out. I then basically proved that would never work. Then Gordon Letwin, who worked for us, came up with another approach where you basically reset the chip all the time and restart it in order to mix the two modes. So, it was kind of problematic. Clearly we were running out of the 640K of memory. And like whenever any address base gets pushed to its limit, you start doing what is called "bank switching" where instead of the address telling you what to choose -- there is a state of the machine where it is choosing different physical memory into the logical address base. And so the 286 looked like an opportunity. We did a version of our form of UNIX called XENIX for the machine. Other people did high-end operating systems. But it just wasn't powerful enough for that. So, the AT ended up being a high-speed DOS machine. And a little bit of a distraction for the industry to figure out what to do with it. We did run Windows on it, but in the final analysis, 286 machines were never really fast enough for Windows. I mean they were nice machines but, during the time of the 286 machine, Windows really did not catch on.

Also, the AT started to show a little bit that IBM's PC group, because it had merged with other groups and gotten very large, wasn't executing perfectly. There were reliability problems. The fact that other people came out with 286 machines so rapidly. They were starting to feel some pressure. When this machine came out they still had 65% market share and that would slide very rapidly over the next five years down to a less than fifteen percent market share for them.

DKA: But, with the 286 you continued your expansion of many new products. Was this a continuation of your strategy? Or, was this a new direction as these products came to the market?

BG: Well, Microsoft always wanted to keep adding business units and do new products. We had started with the system software and languages and we kept going on with that, but we really added a lot in the way of applications. PowerPoint -- we went and bought a company that had pioneered the idea of a presentation piece of software. That was very successful. We got into the mail business. The red box is our Macintosh boxes. We kept enhancing Word and now Excel. Word on the Macintosh was graphical. And then we put Word on Windows. We kept doing new versions of the mouse. You are seeing here that the Mouse is getting a sleeker design than the original Mouse had.

We started to ship Microsoft Windows, a version that was specific to the 286 and later a version for the 386. Now, one of the big controversies of this era was some of the work that we were doing with IBM. We knew that DOS was being stretched. We needed to get to more addressing capability with DOS. And a lot of other features; multitasking, protection, things that DOS didn't have. And so we had started our own project, early on, called DOS 4. And then IBM came along and wanted to play a stronger role than just licensing. That was something we were a little worried about, because of multi-site development and the way their developers were trained was very different from the way we worked. But we came up with a way of trying to work together., The project although during this time was called CP DOS or Big DOS, when it was finally announced, which was part of the next generation after the AT that IBM did, which they called the PS/2 launch in April 1987[laughs].

That was when they announced this operating system OS/2. It actually shipped somewhat[laughs] later than that. So, during a lot of these years we were not only working on Windows, we were also working on OS/2. Depending on how good of a job IBM does in helping promote OS/2, even though it was a lot bigger, we were thinking that might come on. We were the company who really got behind OS/2. We did Excel and Word. We were the first people to do applications for it as it got out there. Also, a graphical application called Presentation Manager came out in 1988 and 1989.

So, there was just a lot of new software development going on at Microsoft, also, a lot more localization doing versions for all the different countries of the world. And taking advantage of the fact that the PC market was not only growing in the U.S., but because the other countries were catching up, it was growing very, very rapidly, particularly in Europe. We were establishing subsidiaries throughout Europe and actually got our European sales up where they were larger than our U.S. sales during this time.

DKA: Bill, we are looking at an early version of Windows. What were some of the major challenges in bringing this technology to the PC?

BG: There are many things that the PC needed in order to be able to run the graphical interface; more memory, more hard disk, more processor speed, and more screen resolution. We started Windows when the screen resolution was 640 by 200. And, the next step up, was a board I did a lot of design on that IBM came out with called the EGA, 640 by 350, that for the first time gave you good character display and graphics. And that was popular for awhile. But it wasn't until another level, 640 by 486, called VGA came out. Likewise in hard disk when Windows started shipping in 1987. We had like ten and 20 megabyte hard disk. Well, it wasn't until they were 80 megabytes until it caught on.

We started with the 286. But it took the 386. Now, at the same time the hardware is getting better we are also improving Windows and convincing people to write Windows applications, so the gestation period was pretty long. But certainly between late 1983 when we announced it and 1990, the primary focus of the company, and the speeches I gave, the promotion I did, was to get people to believe in graphics interface whether it was Macintosh or Windows. And that was a tough thing because people like WordPerfect and Lotus refused to put the resources into doing applications. Just like Dave refused to get involved with Apple's Macintosh. So, it was a serious problem and we just kept having to push and push.

One of the developments that was a major thing is this move to the 386. Unlike the 286, the 386 was designed after the PC had gotten popular. So Intel knew they were designing a chip that was to work with PC software. So, they did a nice job of handling the ability to run the original PC applications, and at the same time, supporting applications that could use lots of memory, the so called, "linear 32-bit address base." In a sense they were just seeing that the Motorola chip was winning a lot of the high-end designs, and taking that linear addressing feature which products like the VAX had had for quite some time, and putting it into their chip architecture in a nice upwards-compatible way.

Now, Intel had promised the 286 and had been very late in getting it done. It had a lot of bugs. Not much availability. A lot of people were wary when the 386 was supposed to come out, and about what they should do about it. Particularly since not much software exploitation done on the 286. And unless you are technical you would think, "Well, if they haven't done the 286, who cares about this 386." Well, to us the 386 was fantastic. And we wanted people to move to it very rapidly so that we could write more powerful software. We had a great relationship with Compaq. And we said to them, well in the same way they had done portables early on, they ought to establish a reputation for high-performance and be the first to bring out a 386 machine. And that was gutsy because they were kind of under IBM's shadow. And maybe IBM when they did a 386 would do it in a different way. But, we convinced them to do it and they did excellent work.

In 1986 they were first to come out with a 386 machine. In fact, that led to a very, very successful period for Compaq, not just in portables, but now in the desktop as well. And that was good news for us because it meant that there was an opportunity for Windows to get in there. We started thinking that OS/2 might really do well because of the 386 had extra speed. In fact, one of the great disagreements between ourselves and IBM was we kept saying, "Let's skip the 286 and do the 386 version and not worry about the 286 version." But they felt like they wanted to put it onto the 286, even though those machines never had enough disk or enough memory. I mean it was certainly in retrospect and definitely our position at the time, that that was a total waste. And that was one of the things that generated some tension between us, as well as the fact, that we were using very small software development teams and at one point they had about 1,400 people working on OS/2 just turning out code that wasn't well integrated.

DKA: One of the famous battles in the history of personal computers, at this time, was the battle, which in some ways is still going on between OS/2 and Windows. What is your perspective on that?

BG: Well, in this period you'd have to call it the "Battle of the Weaklings", because until 1990 neither product sold at all. And in 1990, Windows took off. DOS was dominant. It was a successful product for us. So we could reinvest in doing great new things. Some people like Windows. Maybe ten percent to fifteen percent of users were using it. And they consider themselves sort of an elite group. We, of course, had Excel on Windows. And so people who wanted the greatest spreadsheet would move up to Windows. People who did desktop publishing, PageMaker, and many others wanted to use Windows. But it wasn't a broad scale thing. And even people who used it, would use it for awhile and then they'd exit back to DOS to use other applications.

That is despite the fact that we put so much effort to make it compatible and to allow you to run DOS applications underneath it, particularly on the 386. But that didn't catch on. OS/2, on the other hand, was just too big. I mean we had thought that IBM with their corporate presence could really jam it on people and get people to use it. And that proved to be an illusion. This whole overall strategy that OS/2 is part of called "SAA" or "Systems Applications Architecture", where supposedly everything on the mainframe, all of that complexity, you just made it identical on the PC. Same graphics model, the Presentation Manager. It just didn't catch on the way that IBM had anticipated. OS/2 was a schizophrenic product. We kept saying, "Let's add features. But no let's make it smaller. Let's do the 386 version." But no, IBM made some commitment to some 286 customer. It was a very painful effort--and a very compromised piece of technology as a result.

We then started up a parallel effort to do, what at the time was called OS/2 3.0, starting from scratch, even though we'd give it that same name. And that later became, as IBM and Microsoft went their own way, that became the project that shipped in 1993 as Microsoft Windows NT. But the work was begun on that as early as 1987. Then Cutler came in and the work really started in a big way in 1989.

Windows announced in 1983 and didn't ship for a couple of years after that. Windows 1.0 actually used a combination of Window capabilities that had tiling and overlapping. But the tiling, which was used in some Xerox products, we thought was hard for people to understand. So we went to pure overlapping in Windows 2.0. This is Windows 2.0 here. You can see we weren't using icons at this point. We were, for the first time, letting you run multiple applications at the same time. We could have developed this thing in such a way that we would just add graphics to a single application. But we decided for the first time to have applications share memory, share the processor, and be able to move data back and forth.

So, Windows was more than just graphics. It was a lot of things all coupled together. We could have gone in little half-steps but we just decided to go all the way there at once. This product, Windows 2.0, certainly did a lot better than Windows 1.0 because there were 386s to run it on and some new applications. But it was actually the mid 1990 release, version 3.0, that got Windows into the mainstream. It was the popularity of advanced graphics. The 386 started to come down in price, even though it had come out in 1986. It was 1990 before the price difference was so small it was obvious you should just buy 386 machines. This is when the 286 started to go down very, very rapidly.

DKA: Bill, you are also getting into networking at this point. How did Microsoft respond to the desire of the marketplace to do more networking?

BG: Well, networking was something everyone had been talking about for a long time. Exchanging data electronically whether it be mail, files, or databases. And Ethernet had been there as a standard. IBM had Token Ring. And the prices of the cards came down. But, actually, the percentages of PCs networked was fairly low. And we saw that as an opportunity. We did networking software called MS-Net that ran on MS-DOS. But our first powerful networking product was the LAN Manager product that actually required the OS/2 operating system. This put us into direct competition with Novell, who was really the most successful in those early networking days providing their netware product for file and print sharing.

We positioned our product as running on a general purpose operating system, more flexible, more tied-in with the PC in a very rich way. But they had immense success with the kind of lean performance they had in their product. And, we were a little bit crippled by the slow acceptance of OS/2 that LAN Manager was dependent on, until much later when Windows NT came out.

DKA: Bill, you had said earlier that Microsoft was interested very early on in pushing toward the area of multimedia. It began to be realized in full form in the 486 machine. Tell us about your vision for multimedia and how you pushed it at Microsoft.

BG: The CD-ROM was the first consumer device to use digital storage. And so on a single disk to encode high-quality music they had over a thousand times as much capacity as a floppy disk. And that is a pretty dramatic improvement. I mean it allows you to take things that would have been silly on the floppy disk and ship them. In fact, the CD costs only \$2.00 or so to press. So, without much increase in cost you've got something far better. Now, it means that the PC has to have one of these CD-ROM drives in it. And, of course, the media is read-only so that it is not without trade-offs. But, we saw this early, well before the first CD-ROM conference. That this would make the PC an information device. So that kids who wanted an encyclopedia or sports information or medical information would turn to it, which would really broaden the use of the PC. So, we had a conference and talked about tools and standards. We really got people galvanized around the idea of the CD as a peripheral.

Now, it took a long time for the drive prices to come down. And all that time we were working on applications, including things like this movie review guide called Cinemania. We were working on an encyclopedia which we call Encarta, an art gallery product, a Bookshelf product, a great number of these that provides a lot of information. Just to show you real quickly, if I'm interested in movies, I have Cinemania in the disk here.

I can select movies with a certain star in it. I can select the type of movie. Let's say that I just want comedies and Westerns. I can choose that. Let's say that I just want the ones that were nominated for Academy Awards. Here, I see all of those choices. If there is one that I particularly like, like *The Graduate*, I just move over and click on that. It is now going out to the disk now, finding not only the review from some of the famous review guides, but also, for example, is a picture of Dustin Hoffman. We can go in and look at the Maltin Review and see what he had to say. If we see, in this Maltin Review, the name of somebody like Anne Bancroft and we want to know about her and what other movies she has done, very straightforward.. We could ask to see that list as well as her biography. And, then, if one of those intrigues us, we can go to that. Or, it is very easy to just go back to where we were, the movie, and click on Dustin Hoffman to see the things that he has done. So, the ratings, the information, the ability to know where you can mail order videos, making up a list, it's all right there. And a lot of fun to work with, if you are somebody like myself, who likes movies. So, with thousands of these CDs you'll find things that almost anyone would find a particular interest in.

DKA: It seems to me that with the advent of the CD Microsoft moves yet another stage in its evolution to being not just a company that is dealing with languages and later operating systems, but now with content itself. Did you see that as a new step in your evolution as a corporate direction?

BG: Well, it is certainly an evolution where, when you go from an operating system to a word processor you're already taking that step; the quality of the manual, the tutorial, the spelling dictionary are probably more important or equally important to the code itself. When you move up to something like *Cinemia*, now although there is a technical element of the speed and the interface, making sure we've gone out and licensed the right reviews and have the relationship with those people so that every year it gets updated. We get the material very rapidly. It's almost like publishing. It is taking the word software in a very broad way. And it means hiring in lots of graphic artists, lots of people who are very artistic in how they think. Now, developers are, you know, managing developers and hiring them, there's a lot of the same things about it.. You want to make the jobs really fun jobs. But it does mean that we are bringing together a lot of skills. And it's very much a new frontier, but one that we see as a critical part of our future.

DKA: Another part of your future, Bill, that is also on this machine, is Windows NT. Do you want to switch to that and talk a little bit about how that was a new corporate direction?

BG: Throughout the history of the PC there was always the question of what kind of application could it be trusted for, in terms of speed, capacity, reliability, and richness of the operating system. Good corporations put their databases, their analysis, down onto the PC. And, of course, as the PC has gotten faster and all of these tools have improved, people have been willing to do more and more. But, the most demanding users, certainly into the early 90's, were using mini computers or workstations, not PCs. Even though they knew that the PC was cheaper, more flexible, better tools out there, they just thought they'd run out of gas. And, a part of that was the speed of the chip. But another part was that Windows and DOS just didn't give them all of the advanced capabilities.

And so we started from scratch with a team headed by Dave Cutler to build a very high-end system that would have the same user interface as Microsoft Windows. And that got named "Windows NT", which is short for Windows New Technology. It is from scratch, 32-bit only, portable to other chips, supports a multiprocessor, and compatible with UNIX. A lot of features. Four million new lines of code. So, in the late summer of 1993, Microsoft shipped Windows NT, which we are seeing here right now. And that got us into a lot of corporate accounts. It gave us a great server platform. And really became the first platform that you could say, any application, whether it came from a mainframe, mini, Mac, PC, whatever, it could be run on Windows NT. And Windows NT was such a high-volume system that that would be attractive. So, it is the best of the network server world, the workstation world, the PC world, even the large computer world brought out in a product running on PC hardware.

DKA: We are going to raise an issue about workgroup computing, if you want to comment on that. We were just interested in how you were positioning Microsoft as a company to respond to the needs of workgroup computing.

BG: Well, that word, "workgroup computing", is the idea of people working together. Now initially that meant building a network so that people could share files or share the printer. Then it came to mean sharing mail messages with each other or sharing a database. But now people are moving on and thinking of workflow and bulletin boards where you have large groups of people sharing complex information. Products like Microsoft Mail or other people's mail packages are also moving up to that. We are building network group capability into the operating system so that the idea of sending messages to other people, in any application. It is built right in. So, we see workgroups as something that is really going to push personal computing to new limits. In a sense, it is the killer application of the 1990s.

DKA: Bill, we just spent the last two days looking at pretty phenomenal history that has come from your beginning with a little BASIC tape. We've mostly talked about product development of machines, but at the same time, because of the work that you've done and the work that your company has done, society has transformed. As you look out from the corporate perspective to the world around you, how do you characterize the changes that you've brought into the workplace, into the home. What do you see as the most significant about those changes?

BG: Certainly, the PC has had a major impact. It has brought computing down to a personal level. It's made all types of businesses more efficient and even started to have an impact on education. It has created a great software business, a very innovative and competitive business that is defining new areas. Now, in terms of the promise of the PC, as you go back to our original vision of a PC on every desk and in every home, that is still not achieved. The PC is today, more of a creation tool for documents and spreadsheets. And only as the PC is used more as a communications tool where you find other people with common interests or find products, or find out what is going on through the PC. But certainly will be happening and it really will get out to everyone. So, there is still quite a bit in front of us. Everyone who has been in this industry has had a chance to participate in something very exciting. It is kind of like early steam engines or factories or something. The timing was right for the people who got to do it. And that is a lucky thing for them. Certainly, Microsoft got to play that role and involved a lot of people, and it's been fun. The impact it's there, but compared to what the potential is, it is still quite modest.

DKA: Some people have said, Bill, that a freer flow of information, a greater control of information was instrumental in bringing some of the changes that we saw in the Soviet Union and what we are seeing in China and are seeing in other parts of the world. Do you think that is an overstatement? Or do you think that is a realistic assessment?

BG: Any time we have new forms of communication it changes behavior whether it is political or business or any type of behavior. Radio and T.V. did that. The PC will be classed as big or bigger an advancement in communications than those devices were. So far it is mostly electronic mail, printing, newsletters. But even at this stage that means that it is so much harder to suppress word getting out about things. You know printing presses used to be big, expensive, and easy to locate. And now, anytime you let people have PCs you are letting them have a printing press that can do very high-quality work and can send electronic messages around with great ease across national borders and to people all over the world. So, certainly, it is a tool that will change politics amongst other fields.

DKA: You mentioned your vision of where the PC will be on every desk and in every home. You clearly have had a vision about the kinds of products that would come out and yet you said a minute ago, "This is just the beginning." What do you see as lying ahead in terms of further unfolding of the vision that you have held onto so continuously over the last 20 years?

BG: Well, the PC will continue to evolve. In fact, you'll think of it simply as a flat screen that will range from a wallet size device to a notebook, to a desktop, to a wall. And besides the size of the screen, the only other characteristic will be whether it is wired to an optic fiber or operating over a wireless connection. And those computers will be everywhere. You can find other people who have things that are in common. You can post messages. You can watch shows. The flexibility that this will provide is really quite incredible. And already there is the mania in discussing this so called "Information Highway" which is the idea of connecting up these devices not only in business, but in home, and making sure that video feeds work very well across these new networks. So we've only come a small way. We haven't changed the way that markets are organized. We haven't changed the way people educate themselves, or socialize, or express their political opinions, in nearly the way that we will over the next ten years. And so the software is going to have to lead the way and provide the kind of ease of use, security, and richness that those applications demand.

DKA: Talking about some major changes, is there anything that is lying ahead that you find scary as well as exciting?

BG: Well, if you look out far enough the computer will eventually learn to reason in somewhat the same way that humans do, so called "artificial intelligence". If you take that far enough you can imagine an evolution, essentially, moving over to silicon-based life and carbon-based life, playing a much more limited role than it does today[laughs]. Now that is in some ways off in perhaps the progress that there will only come through the cross effect of the sequencing and understanding of human genome and the learning techniques that have evolved for us and then mapping those into a substrate that will have better execution, that is, instead of chemical transmittal, the high-speed electronic implementation of some of those same algorithms. So, that could be a little scary. In the meantime, which is a very long time, the computer is a great tool and it is a very empowering tool and there is a lot of fun work to be done.

DKA: Bill, we've talked about many of the continuities in Microsoft and some changes in Microsoft. Do you think the corporation you have will continue down the path, or do you see some major new branches that you'll be heading towards in the next few years?

BG: Every three years are important in terms of redefining what we do. Any company that stays the same will be passed by very quickly and there are lots of fine examples of that. So, working with content companies, phone companies, cable companies, thinking through even radically more user easy-to-use interface that we've had to date. These are all necessary things. And because we now have a research group, and we are out there working with lots of universities and are able to continue to hire great people, I'm very optimistic about our future. But, it is a future full of change and surprise.

DKA: Is there anything else you want to add?

BG: I'd say that my job, throughout all this, has been, I think, the most fun job I can imagine having. And partly the people I've gotten to work with outside the company. Certainly there are great people inside the company. And certainly, for at least a decade or [laughs], that will just continue to be the case.

DKA: It has been a terrific pleasure. You've worked very hard and I thank you so much for your help.

BG: Sure. Someday I will be glad I did it.