IN THE UNITED STATES DISTRICT COURT

FOR THE DISTRICT OF UTAH, CENTRAL DIVISION

NOVELL, INC.,)

Plaintiff,)

vs.) Case No. 2:04-CV-1045 JFM

MICROSOFT CORPORATION,)

Defendant.)

_____)

BEFORE THE HONORABLE J. FREDERICK MOTZ

DATE: NOVEMBER 21, 2011

REPORTER'S TRANSCRIPT OF PROCEEDINGS

JURY TRIAL

VOLUME XVIII

REPORTED BY: Patti Walker, CSR, RPR, CP

350 South Main Street, #146, Salt Lake City, Utah 84101

A P P E A R A N C E S

For Plaintiff:

DICKSTEIN SHAPIRO
BY: Paul R. Taskier
Jeffrey M. Johnson
Miriam R. Vishio
1825 Eye Street N.W.
Washington, D.D. 20006

WILLIAMS & CONNOLLY
BY: John E. Schmidtlein
725 Twelfth Street N.W.
Washington, D.C. 20006

SNOW CHRISTENSEN & MARTINEAU
BY: Max D. Wheeler
10 Exchange Place, 11th Floor
Salt Lake City, Utah 84111

NOVELL, INC.
BY: Jim F. Lundberg
1800 South Novell Place, H-544
Provo, Utah 84606

For Defendant:

SULLIVAN & CROMWELL
BY: David B. Tulchin
Steven L. Holley
Sharon L. Nelles
125 Broad Street
New York, New York 10004

MICROSOFT CORPORATION
BY: Steve Aeschbacher
One Microsoft Way
Redmond, Washington 98052

RAY QUINNEY & NEBEKER
BY: James S. Jardine
36 South State Street, #140
Salt Lake City, Utah 84111

I N D E X

Witness	Examination By	PAGE

SALT LAKE CITY, UTAH; MONDAY, NOVEMBER 21, 2011; 8:00 A.M.

PROCEEDINGS

THE COURT: Good morning. Stay standing for the jury.

(Jury present)

THE COURT: Good morning, everybody. I hope you had a nice extended weekend. As you know, we'll be sitting two days this week. We will not be sitting Wednesday through Thanksgiving.

Yes, Mr. Holley.

MR. HOLLEY: Good morning, Your Honor.

Microsoft calls Bill Gates to the stand.

BILL GATES,

Having been duly sworn, was examined

and testified as follows:

DIRECT EXAMINATION

BY MR. HOLLEY:

Q Good morning, Mr. Gates.

What was your first exposure to computers?

A When I was in the eighth grade at age 13, the high school I went to, Lakeside, got a terminal that was connected up by a phone line to a large expensive computer.

Q I would like to put up on the board demonstrative Exhibit 171. I'm going to show you a copy, although I think you have the screen there in front of you.

What is this picture?
A That machine is how we communicated with the big computer. It's called the teletype. Paul Allen, the co-founder of Microsoft, is seated and I'm standing looking over him while we're trying out computer programs being run by this large machine that was about a thousand miles away.
Q Was there any particular programming language that you used back in 1968 in the eighth grade?
A Yes. The machine we connected up to was a General Electric machine that had a high-level language called BASIC, which stands for beginners algebraic symbolic instruction code. So it was a way that you could write 20 or 30 lines of code and get a program like a tic-tac-toe.
Q Mr. Gates, something seems to be wrong with that microphone. There's a lot of feedback. Is there --
What got you interested in the idea of computers for use by individuals, by normal people?
A Well, Paul and I loved working on computers and took every opportunity we could to learn about writing software, and we decided that if they were cheap enough, that it would be great not only for us to have one that we could use ourselves, but for everyone, keeping track of your schedule, your information, trying out different experiments. So we were fascinated by the idea of would the price of these things ever come down, where was the technology going, and

would it be pervasive.

Q Was there any particular product that came on the market that captured your attention in that regard?

A Absolutely. This GE computer, if you had seen it, would have filled half of this courtroom. It was gigantic and it had vacuum tubes, and just keeping it running required many full-time technicians. And by today's standards, it was about a 10,000 power that you have in a phone -- a typical phone that you might carry in your pocket now.

So the thing that allowed us to go from that big machine to where we are today was after taking the logic of the computer and putting it on to a chip. And so Paul saw an article in Electronics Magazine all the way back in 1970 that talked about these chips and the fact that every couple of years they could put twice as many elements, transistors, on to these things. And that kind of exponential improvement meant that the equivalent of this machine that we were using would eventually be very inexpensive.

So the only missing element to this idea of this wonderful tool was to have a lot of great software that would run on it. So it was the invention of the chip, one came out in 1971, an early computer run on a chip, and Paul said to me can you write a BASIC for that, because I had written BASIC Interpreters for some of the more expensive

machines. I said, no, it's not good enough.

But then in 1973, a far better one called the 8080 from Intel came out, and Paul got me the manual for that, showed it to me, and I said, absolutely, I could write a BASIC Interpreter. So we thought, wow, now is the time that people can have personal computers.

Q Was there any particular brand of computer running the Intel 8080 chip that you and Paul Allen focused on?

A Well, what happened was that chip came out in early 1973 from Intel. And then I went -- I started college back in Massachusetts at Harvard, and Paul wanted to brainstorm more about, okay, what should we do in software. So he moved out there and sort of took a leave. He was two years ahead of me. So he took a leave from Washington State where he had been going to school. And we were talking about when is this going to happen, it's powerful enough.

And then we saw on the cover of Popular Electronics Magazine a kit computer based on this chip that had come out early in the year called -- the kit computer was called the Altair 8800. When we saw that, we thought, wow, this is fantastic. We also thought, wow, this could happen without us. We want to write our BASIC and write lots of software. And so that was a key milestone was the scene that that Altair 8800 had come on to the market.

Q I would like to show you what's been marked as

demonstrative Exhibit 170.

What is this a picture of?

A On the left you have the issue of Popular Electronics Magazine that we saw on a very cold wintery day in Boston, and you can see it says on there world's first mini computer, because the term personal computer essentially didn't exist. The machines that were like $10,000 to $100,000 were called mini computers. The big computers were called mainframes. So you had mainframes, which were like the one I originally used, and these mini computers. These eventually became called personal computers or microcomputers. But anyway, it's giving the machine too much credit to call it a mini computer, but that's what they did.

Then on the right you see -- my right -- the actual machine itself. The way you actually entered a program in is you took those switches and you flipped the binary machine code -- this is very low-level stuff -- in by moving the switches a lot. And the only -- unless you attach something to this, you just bought this, which was about $400. Then all you could do is make those lights flash.

Now if you hooked up one of those teletypes which we saw earlier in the previous photo, then you could type into it -- if you had entered the right software, you could communicate with that teletype.

So this is the very first personal computer. Its power is so limited that some people think, no, maybe you shouldn't label it the first personal computer. But for me this was the first idea you bought for the individual and it was reasonably priced.

Q Mr. Gates, what, if anything, did you and Mr. Allen write for the Altair 8800?

A This computer, of course, had that Intel 8080 chip inside it. And so unless you had a piece of software that helped you, you had to learn this 8080 machine code and key it in.

What we decided is to write a BASIC Interpreter, that is the software that would let other people write BASIC programs, and we would put that onto a paper tape which you could feed into that teletype. And so once that was all loaded up, then you would be able to type in BASIC programs and run them on this Altair computer.

This thing had such limited memory, we had to write the smallest BASIC Interpreter ever written -- which I loved writing small, really compact pieces of software. So even on this machine with 14 bytes, we made it so our BASIC Interpreter could run very comfortably.

Q Was your BASIC Interpreter ever sold?

A Yes. The start of Microsoft is when we call up these people and say, hey, we would like to write software for

your machine, and they kind of said, yeah, a lot of people are calling us, what do you really have. Well, we have nothing. So we went to work. Paul had a brilliant idea that we could simulate this machine on a big computer. And so on a very large computer that we had access to on the Harvard University campus, we simulated this machine and wrote the BASIC Interpreter. So it took us about four months.

And then we printed out a paper tape and we called them up and we said, how do you get -- how do you communicate with the teletype, how do you get the characters, what's the input, output, how does that work. They said, well, you must be serious because you're the first one that's asked us that question. Let me go get the guy who knows. And so they explained that to us. So we made it set up.

So Paul flew out to Albuquerque with this paper tape hoping that we had gotten the simulator right. They had a machine there with this teletype in the computer, and Paul loaded up the software, and amazingly it ran, and it was incredible.

So they licensed -- to answer the question, they licensed the BASIC Interpreter from us and became the very first customer of Microsoft. And they went on to sell tens of thousands of copies of that BASIC Interpreter, which gave us the money to hire our friends and write a lot more

software.
Q How old were you, Mr. Gates, when Microsoft was founded?
A I was 19 when we actually put it together, which meant that it kind of messed up my college career. But it -- you know, it was -- we wanted to be there and be the first.
Q Where did the name Microsoft come from?
A Well, we didn't want to call ourselves like Gates and Allen because we had this idea that eventually this could be a big company. And so we took micro from microcomputer, which was one way of describing these machines, because they were based on microprocessors. That's what the chips were called. And then we thought the key element that would unlock and make these things fun to use and valuable to use was software. So we thought software from microcomputers. We decided we didn't like soft micro. So we went with Microsoft. There were no other companies, so nobody had taken that name, even though it was kind of an obvious name.
Q Did you have a vision for Microsoft at the time the company was founded?
A Well, we had a vision for the industry, which was that even though we didn't make personal computers, we thought eventually they would be interesting enough that we would have a personal computer on every desk in every home. That was literally the slogan was we were going to write the

software that would make it worthwhile to have one of these computers on every desk and in every home.

Q Was the idea of having a personal computer on every desk in every home a common perception in the mid 1970s?

A No. It's hard to go back in context of those days. Even the person -- the computer company we admired the most with whom Paul and I had done work for in high school was this Digital Equipment. And the CEO of that company, Ken Olsen, said he couldn't imagine anybody would ever want a personal computer. And this from the guy who, you know, had been a visionary, essentially created the mini computer business that was underneath the mainframe business, he thought, well, come on, what would they possibly do with it.

So, in fact, it ended up being largely a group of fairly young, naive people who saw this dream of personal computing and started up the hardware companies and software companies that have eventually led to where we are today to where it's sort of a commonplace idea that these are great tools for everyone.

Q How long were you a full-time employee of what became the Microsoft Corporation?

A Well, by -- so we write the BASIC in early 1975. And then I go back to Harvard, go and work with Microsoft, go back to Harvard one more time, so I actually got three years of school done. By June of '76, things were busy enough

that I never did go back. I'm still officially on leave. I was close. I mean literally I got three years more than your average dropout has done.

But anyway, from June of '76, which is the last time I was at school, until January 2000, which is when I turned over the CEO role to Steve Ballmer, I was the CEO of Microsoft.

Q Did you assume a different role in 2000?

A Yes. From January 2000 until June 2008, I was still very full-time at the company, and we called my title -- it's kind of an unusual one, it was chief software architect. And so I got to give up a lot of stuff about sales and personnel type things and focus more of my time on software architecture, Microsoft research, the more advanced things we were doing in terms of our engineering plans.

Q You mentioned Microsoft research. Can you tell us what you mean by that, sir?

A Well, Microsoft, you know, we loved hiring engineers. So as the products were more and more successful, we hired lots and lots of engineers. And eventually we got to the point where we thought we want some of these engineers to take a very long-term view of what is possible in software.

And so we formalized the idea that the product divisions would have about a four-year time frame in their work, and we created this new division called the research

division that would have things like computer vision, computer speech, proving correctness, anyway, very hard and long-term type things, and we put that into the research group. That got kicked off in about -- actually they're celebrating their 20th anniversary. So that got kicked off, that formalization of the really advanced stuff, in 1991.

Q Does all the research and development that occurs at Microsoft occur within Microsoft research?

A No, that's just the far out stuff. So the bulk of the engineers are in the actual product divisions, so the Windows division, the Office division, the sequel database division. Nowadays we have a search product called Bing that competes with Google. There's a big division that does that.

So when you speak about engineering at Microsoft, it's research and development, and the product groups as a whole are actually by far the largest part of that headcount.

Q Over the last ten years, on an annual basis, can you give the jury some sense of what the scale of Microsoft's research and development expenditures are per year?

A Yeah. Our R and D spending has been well over eight billion in each of the last five years. Microsoft is the -- has a larger R and D budget than any company in the world.

Q Why does Microsoft spend so much money on research and development?

A Well, the belief is that software can keep improving, that is that even though we have some pretty good software from us and other companies today, that we're nowhere near even some of the things that Paul Allen and I talked about so many years ago. You know, you still can't talk to computers and have them understand. They can't translate speech. You can't take your handwriting and just scribble it on a little tablet like thing and it knows what you want.

You know, software is still kind of plunky. It's the human accommodating it instead of the other way around. So the idea is that in each of our products areas and in these advanced research areas, that by spending all that money, we'll come up with much improved products and that the sales -- not only will that be a good thing in terms of the research, but that there will be an opportunity to sell those products and actually make back more than the billions we invest.

Q Mr. Gates, what have you been doing since you retired as the chief software architect from Microsoft in June of 2008?

A Well, at that time I changed my focus to the foundation, the Bill and Melinda Gates Foundation, that my wife and I -- my wife Melinda and I run. I'm still part-time at Microsoft, but my primary full-time activity is the work of the foundation.

Q Can you tell us what your personal principal focus is, what you do at the foundation?
A Well, the two biggest things we're trying to change at the foundation, the first is health, where there's still a lot of diseases like malaria, TB and AIDS that we haven't invented a vaccine for. So the idea is to -- you know, we've put -- spend billions a year literally on backing scientists who can come up with these new interventions. Then we also spend on making sure those things get delivered into the poorest countries where those diseases are a huge issue.

The other big cause for us is U.S. education, and are there ways to help teachers learn from each other in a better way or use technology in a better way. So most of my time is either on those global health issues or the work related to U.S. education.

Q Mr. Gates, you said you were still part-time at Microsoft. Can you tell us what your role is on an ongoing basis at Microsoft?
A Well, I'm called chairman of the board, and I would say there's two things I do, although it's only about ten to 15 percent of my time. One is that, as a board member, we have regular meetings, strategy retreats, you know, discussions, phone calls about big decisions the company is making, like an acquisition or a big personnel change.

The other thing is that there are a few product groups that still like to meet with me and talk about their plans. So, for example, the Bing group is one that I meet with on a pretty regular basis, some people in the Office group. So there are a few of the product strategies I keep my hand in and provide advice, but I'm not full-time, so there is a lot going on that I'm not up to date on.

Q Is there anything you miss about being at the company on a day-to-day basis?

A Well, I love my work at Microsoft, to work with smart people in a fast moving industry that's, you know, full of great competitors who can do surprising things and do something based on, you know, trying to figure out what customers are going to like. It's a wonderful thing, and many of the dreams about what software could do are still out there in front of us. And so I miss the opportunity to do those things.

I have to say, though, that the same kind of excitement I get, although a slightly different flavor, by working with scientists on these vaccines related to the new diseases, or technology to help kids learn.

Q Mr. Gates, I would like to change topics now a little bit and talk about the operating system business. When did Microsoft first become involved in the operating system business?

A The very earliest personal computers ran Microsoft BASIC, so it operated like an operating system. So the Apple too, when it run our BASIC Interpreter, the TRS80, which is the Radio Shack machine, all those early machines, 8-bit machines, the BASIC was the thing that ran on the system.

Then later as people got disks, even some of those 8-bit machines, there were some general operating systems worked on. But when we moved up to 16 bit, which was the permanent computer, IBM was one of the first people to do one of those more advanced machines. That's when Microsoft created what you would call a classic operating system. It was actually during the discussions with IBM about where we were encouraging IBM not to do an 8-bit machine. But to be one of the first to do a 16-bit machine, we went out and licensed some work, which we later bought and enhanced, and that became Microsoft's first operating system product that was called MS DOS.

Q Was MS DOS a product that ran on the IBM PC?

A Yes. The IBM -- if you bought the original IBM PC, it didn't come necessarily with a disk. So when you turned it on, it just ran the BASIC Interpreter. When you bought a disk -- which, or course, nowadays all of them come with disks -- then you would plug in a diskette and run the MS DOS, although IBM relabeled it the PT-DOS operating system.

So all the disk-based machines became -- could run an operating system and most of those users chose to run MS DOS.

Q Was MS DOS available exclusively on IBM PCs?

A No. Part of the beauty of MS DOS was that it ran the same on over 50 different types of hardware. And so both as a user in terms of learning the funny instructions, which now seem very arcane, but the instructions for how you ran the disk operating system and for people developing software, that uniformity that you could pick whatever hardware you wanted from Wang, or TI, or IBM, or Hewlett Packard, Compaq, you could pick any hardware, yet when you put DOS in, it was this familiar DOS prompt, and you could go out and buy in a store any of the MS DOS applications, which were not just -- eventually as we did a great job encouraging more and more applications to be written, the breadth of those got up to the tens of thousands of applications, you could go and buy those and plug it into your computer knowing that if it had been tested against MS DOS, it would run on that machine.

Q What benefit, if any, did the fact that MS DOS ran on different brands of personal computers have for independent software vendors, or ISVs?

A Well, this idea that the hardware innovation and software application innovation could proceed independently,

that is somebody could come out with a better piece of hardware because it ran MS DOS and you didn't have to rewrite all the software, or some application software company could write a new application and that entire base of MS DOS machines would be available to them, it made it a lot easier for them to decide to invest in R and D.

Software is a very -- has very high fixed costs, that is to write the very first copy of a software product, you have to pay the engineers, take a lot of time and get it done. Now once you have that product done, your extra costs to make additional copies, the marginal cost is very low, I mean almost is the cost of the media.

So if you want to sell things at a low price that, say, an individual or a student would be willing to pay, or a small business, then how do you make it work? Well, the only way to make it work is volume. So the fact that MS DOS was out there and lots of volume meant that you could put a lot more into R and D, price the product to sell, and yet know that the volume would be available to you, so the very fact that -- the reason the software industry was able to bootstrap -- to get going, to go from nonexistent, when Microsoft went into business, to this gigantic business was because of this volume thing. As there were more machines meant you could write more software. As you wrote more software, the product industry, it gave people more reasons

to buy machines. So the volume of the machines and the amount of good software, those two things fed on each other in this very positive way.

So now, you know, we have lots and lots of software companies, including very specialized software, you know, for different types of farmers or funeral homes or, you know, you name it, there's a lot of software out there for most applications.

Q Starting with MS DOS back in the 1980s, did Microsoft choose to write applications to run on its operating systems?

A Yeah, absolutely. I don't know of any case -- anybody who created an operating system that didn't also do applications to run on their operating system. You know, in our case, we did a Multiplan spreadsheet, a Word word processor.

There's also a class of software that you do called tools that run on the operating system that are there to help people develop software, so to make it easy for them to write software. There was quite a variety of software tools you would want to have for lots of ways of doing development, including what are called high-level language compilers and interpreters. So Microsoft did a bunch of tools and a bunch of applications to run on MS DOS.

Q The jury has heard references to something called

Visual C++. Is that a tool, the kind you're describing?
A Right. C is a mid-level computer language, much better than machine language, but not super high level, and it's used for a lot of applications developed in C. Then there's a sort of fancy variance of it called C++. Then Microsoft's specific tool that helps you develop in that was named Visual C++. So that's a C compiler product, a tool for development that actually itself ran in MS DOS, but whose goal was to help you write originally MS DOS and then later also Windows-based programs.
Q Did Microsoft create Visual C++ only for its developers?
A No. We sold hundreds of thousands of copies of Visual C++. So it was used by anybody who was going to write in C, you know, that became the most popular tool to do that. So both developers doing commercial applications to sell to each other and then also corporate developers where inside a company to write some special applications just for your own company, that's actually a huge market, a lot of developers there. So Visual C++ was immensely popular in all those different types of developers.
Q Mr. Gates, what benefits, if any, did Microsoft derive -- Microsoft itself derive from both creating operating systems and the applications that run on top of those operating systems?

A Well, by writing applications, you kind of show confidence in your own operating system, you show that, hey, I really believe this is going to catch on. If people think you have good software developers, it's all the more reason for them to take a look and say, you know, should they also write for.

Also, when it comes to figuring out what new things you want to add to the operating system, what needs to be sped up, what features people would take advantage of, having the people who are doing these applications, are sometimes called first-party applications, is a -- it's a huge benefit.

Now your applications people, they have to, you know, do a lot of work to help you pioneer things. They get stuff when it's still particularly buggy and not quite right. You know, sometimes you do go down dead ends, but, you know, they are part of the process of figuring out where you want to go with it.

Q How, if at all, do these independent software vendors, or ISVs, benefit from the fact that Microsoft developed both operating systems and the applications that run on top of its operating systems?

A Well, there's a huge plus in terms of the maturity of what we come to them and show them that it's going to be made available. In many cases their applications are

complimentary to our applications. Now if they compete directly with our applications, that's certainly a minus to them. But the broad benefit is that the knowledge of what should be in the operating system, the sophistication of the operating system is greatly enhanced by having an application division.

Q Are there any instances in which technologies developed by Microsoft's applications developers have migrated into the operating system?

A Yeah, absolutely. What you do when you're doing a new version of an operating system is you take a hard look at things that are showing up in many applications. So if they are all having to write printer drivers, for example, and that's lot of trouble for them, you think, well, maybe the operating system should do the printer drivers. That example is a good one because MS DOS does not have printer drivers.

So all the word processors, the Microsoft word processor, the WordPerfect word processor, MultiMate, MicroPro, they all spent a huge amount of time writing printer drivers. When we moved up to Windows, we took and we put the printer drivers in. So there are many times where you'll see what applications developers are having a tough time on or what's showing up in many of these applications and you will decide, okay, that's the next

thing that the operating system should take on so that it's easy for applications, instead of doing it themselves, to simply call that functionality knowing it will be better in the operating system. So there are hundreds of examples.

Another good one is that handling different languages worldwide, like Arabic goes right to left, and Japanese, Chinese and Korean have these characters that are large, and there's like 30,000 kanji, which is Japanese, initially all the complexity of dealing with world languages were in the applications. And so most applications didn't get Arabic correct or they didn't get Chinese correct.

Now over time, we saw that that was showing up in the applications and we moved that down into Windows. So nowadays writing an application that works with all the world's languages is a hundred times easier than it was before because that moved from being an application level feature to an operating system feature.

Q I would like to take you back to 1984 and the state of Microsoft at that time. Was MS DOS a success in 1984?

A Yes. By -- MS DOS comes out in 1981 with early 16-bit personal computers, like the IBM PC, the Compaq, and quite a few others. By 1984, there were a lot of applications being written for MS DOS, a lot of new hardware coming along. So, you know, it was clearly in a good position.

Q I would like to show you what's been marked as

demonstrative Exhibit 173.

You have different glasses, Mr. Gates, but how old were you when you were on the cover of Time Magazines in 1984?

A Let's see, I must have been 28 -- 29. Sorry.

Q Twenty-nine. Okay.

Were you content at this point to rely on the success that Microsoft was having with MS DOS?

A No. In fact, even in the article that's inside that magazine, we're talking about how the next frontier is a much richer interface that includes running multiple applications at the same time, applications working together in rich ways, and presenting a graphical type appearance on the screen. So we were already hard at work on what was the next generation of operating system capability.

Q Now you mentioned graphical interfaces. Did MS DOS have one of those?

A No, that became a key distinction. MS DOS only ran one application at a time and it would take over the entire screen. The next big change, which for us we called Windows, was where we ran multiple applications at a time and they could share the screen, hence the name Windows.

But perhaps most dramatically what was in those Windows, instead of just being mono sized fonts, just kind of boring characters, you could draw any type of picture. In fact, it was lots of little individual points, pixels,

but because of that, it was very flexible, so the text became much richer with fonts and variable spacing. And the interface, now you could do essentially anything. You could draw a picture basically. And so a whole new set of interfaces became possible.

By 1984, we had oriented the company to put most of our effort into building the operating system what became Windows 1.0 that shipped in 1985, and starting to build applications that worked that way both in our case for Windows that ran on the Intel-based PCs and then also for the Macintosh that ran on Apple hardware that used a different microprocessor.

Q Now I would like to --

MR. HOLLEY: And I'm going to hope that they can do this for me. Can we see both 176 and 179 at the same time on the screen? I should have asked this before.

THE WITNESS: That's pretty good.

He's got Windows.

MR. HOLLEY: Yes, he does.

BY MR. HOLLEY:

Q Mr. Gates, the jury has heard a bit about the difference between character based and graphical user interfaces, and we don't have time for an in-depth lecture, but can you just at a high level explain what in your view was an improvement in terms of the graphical user interfaces

that's on the right-hand side of the screen?

A Yeah. I think just glancing at this will give people a pretty clear understanding. On the left there, that is MS DOS and that thing called the C prompt, C with that greater than sign, that was the interface of MS DOS. So somebody there typed DIR and they see their file names here, the names and the sizes. And then once it had done that, it would then -- then you got another C prompt, and you could type -- if you typed in the name of an application, it would run, but it would only one run one at a time. It would take over the screen. So that was pretty limited.

Then over here, that's actually a screen shot. It looks like it's a Win95 screen shot. You can see that now instead of just text, you have graphics. These are called icons, so these little pictures associated with actions you could take. And so inbox is something you would click on to see your mail. Internet would run a browser that could let you look at the Internet.

Here if you clicked on this upper left one, my computer, it shows you what is underneath -- what's in my computer. I have a floppy drive, a CD ROM drive, a hard disk drive, and a printer. So these -- as you open it up, this is called a graphical folder. It shows you the set of things inside. So as you click, you see what's inside.

If you click on printers there, it would show you the

names of the printers. Usually you would only have one, but you would get a folder like this that would have that name in it.

So this interface is very different. Actually, you know, you have to get used to it. But also it is harder to write applications for graphic interfaces. It takes quite a bit of adjustment because you've been used to writing to this thing on the left. The complexity and skill set for writing to the one on the right is quite a bit different. And the user, it takes a little while to adjust. Believe or not, some people really have gotten use to this thing on the left. Once you get used to the thing on the right, you would never go back.

Q Mr. Gates, you said in the mid 1980s that Microsoft was refocusing on graphical interfaces. Was that a universally accepted view at that time in the mid '80s that graphical interfaces were the way to go?

A No, absolutely not. It was considered crazy even by the majority of the industry. The machines were slow enough that Windows 1.0 that we shipped in 1985, which was a brilliant piece of work, I mean we tuned it and tuned it and tuned it, it was late -- quite late, because we were tuning it. Some features didn't work out. We kept changing it. But even with our best work, it was pretty slow.

And, remember, you know, you had floppy disks back

then. So it took seven floppy disks you had to stick into this thing to install it. So just to get Windows 1.0 up and running would take you a long time. So people thought we were kind of nutty to be going in this direction.

Now fortunately for us, Apple also had bet on graphics interface in their creation of the Macintosh computer. So the two companies that were sort of pushing the most in this direction were Apple and Microsoft.

IBM, for example, thought, you know, it was wrong, and they went off and did a different thing with this character mode -- fancy character mode, character mode multitasking with Windows, but character mode Windows, they did a thing called Pop View.

Q Now you used a term that I'm not sure we've heard much in this trial, which is multitasking. Can you explain what you mean by that?

A Yeah. That's just the idea that can you run more than one thing at a time. So DOS, using this terminology, would be single tasking, one task at a time. Whereas Windows, or Unix -- nowadays it's hard to think of an operating system that's single tasking. Most operating systems now are multitasking, so you can run multiple things. And so even as a user, you can start up in word processing and be working along and then think, okay, I want to go over to some application, go work on that. You can even have them

share the screen with each other. So multitasking just means multiple actions at the same time.

Q Did Microsoft attempt to persuade third-party developers or independent software vendors to write applications to early versions of Windows?

A Yes. Starting actually back in 1983 when we decided to go in this direction, we hadn't even named Windows. Windows was called interface manager at the time. We started to talk to software developers. We intensified that. Of course, in 1984, Windows 1.0 was finally getting done. Then Windows 1.0 came out. We went out and said, hey, this is great. There were several software companies who bet with us.

Then our own applications, you know, by 1987, '88, '89, we're getting our own applications, Word, Multiplan, out in the Windows environment. So there starts to be some momentum, but it was still tough. Certainly until sometime about 1990 or 1991, there was still some major holdouts who didn't think graphics interface was the way the industry should go.

Q Were there any particularly significant ISVs that were holdouts, in your terminology?

A Well, some of the biggest didn't get started very early, and even when they did get started, they had put sort of the second string, not as well staffed --

MR. JOHNSON: Objection, Your Honor, to whether or not they put the second string on. This man doesn't have personal knowledge of that.

THE COURT: Overruled.

Go ahead.

THE WITNESS: Well, I had an opportunity to meet the people, the people they were assigning to their Windows development work in many cases. And, you know, some people were more serious about it than others. Some of the major companies, including Lotus and WordPerfect, were not as serious, didn't see it as a big thing that would happen and be critical.

BY MR. HOLLEY:

Q I think you said that Windows 1.0 was released in 1985. Was that a commercial success?

A Not really. I mean it would lead to a commercial success, but that version itself did not sell a lot of copies. You know, in fact, we -- you know, we made it really easy for hardware people to install it. We actually came up with a version that you could ship with applications. We did a lot of things. I mean we sold hundreds of thousands of copies, but the percentage of machines that ran it was a very small percentage.

Now Windows 2.0 picked up that number a fair bit. Then Windows 3.0, the numbers started to be quite significant.

That really was like about 1990. That's a big milestone because you have everything progressing in parallel. You have the size of the memory going up, the size of the disk going up, the speed of the processor, the resolution of the graphics. You have all these things that are favorable to graphics interface, including printers. You know, old printers were literally character printers. Then finally you have these matrix printers, and then eventually laser printers. So you have a lot of things that made graphics, including not just Windows, but also Apple's promotion of Macintosh, a lot of things that are slowly but surely driving graphics interface to be this fairly mainstream activity.

Q You mentioned a couple of times the Apple Macintosh. What, if any, products did Microsoft develop for the Apple Macintosh?

A Microsoft actually put more people on to the Macintosh project than Apple did. We did five different pieces of software for the Macintosh. You know, we thought it was quite an exciting machine. We did a word processor, Word. We did a spreadsheet that was first Multiplan and then Excel. We did a database program called File. We did a BASIC Interpreter. We did a charting package that also was eventually replaced by Excel.

Q Were the versions of Microsoft's applications for the

Apple Macintosh successful in the marketplace?
A Yeah, they were very successful. Those became still to this day the best selling applications on the Macintosh.
Q Did Apple itself develop applications to run on the Macintosh?
A Yeah. They did a word processor called Write. They did a draw package. They did a paint package that they actually included for free with the machine, that they bundled with the machine. But they created a whole division called Clarice that sold their applications, including their word processing application, their drawing application, and eventually a whole bunch of other applications.
Q Were the Clarice applications written by Apple more popular with consumers than Microsoft's applications?
A No. Microsoft succeeded in having the best selling applications in the categories we were both in, like word processing. And overall our dollar sales, unit sales were higher, which was a great thing for Apple, that it helped popularize the Macintosh.
Q I just want to be clear I understood your answer. The success of Microsoft's applications was a good thing for Apple?
A That's right.
MR. JOHNSON: Objection, leading.
THE COURT: It was. Overruled.

You can answer.

THE WITNESS: Okay. Yes. The more applications you have that show off, particularly if they are well adapted to take advantage of the unique features of the machine so they show off what that machine can do, the more really good mainstream applications you have that are well tailored to your latest hardware and software work, the more attractive that machine is to people who are considering buying it.

BY MR. HOLLEY:

Q Did Microsoft have more or less access to information about the Macintosh operating system than Apple's own application developers had?

A At the very early stage of the Mac project where we were the only software developer outside of Apple, we had this very close relationship and even helped define some of the user interface, the very first disk format, which was a piece of work that I did. But then once the Macintosh shipped and they understood how serious we were about Windows, they decided to --

MR. JOHNSON: Objection to what they decided, Your Honor.

THE COURT: Yeah. That's probably hearsay.

Go ahead.

MR. JOHNSON: Thank you, Your Honor.

THE COURT: What did you do in response? I mean the problem is -- tell us how you know what they knew.

THE WITNESS: Well, Apple decided by -- and I know that because of Apple saying they had decided --

THE COURT: Go ahead.

MR. JOHNSON: That's the problem, Your Honor.

THE COURT: It's also the problem he's obviously doing things in the market in response to what Apple was doing.

Go ahead and answer the question.

THE WITNESS: Well, I had conversation with Apple employees where they say we --

MR. JOHNSON: Your Honor.

THE COURT: The problem is that's hearsay because they told --

MR. HOLLEY: Your Honor, let me ask a question and I think we can cut through this.

BY MR. HOLLEY:

Q How, if at all, did the amount of information that Microsoft received from Apple change after Apple understood that Microsoft was developing Windows?

A Well, Apple chose not to give us access to much operating system information, including they were more generous to other third parties with that information than they were with us. A lot of times we would find out about

features literally when the operating system shipped to end users.

Q How, if at all, did writing applications for the Apple Macintosh help Microsoft write applications for Windows?

A Well, as part of the whole thing of learning how to write graphical applications, how should they look like, what are you going to do in the dialog boxes, how are you going to make them fairly efficient, we shared over 80 percent of the code that we wrote in those Macintosh applications which shared in common with the versions that we wrote for Windows. So the base code was largely the same, although there were significant pieces that had to be adapted specifically for Mac or specifically for Windows.

So doing the Mac work was the key part of the learning curve. In fact, we were the most committed ISV to the Macintosh partly because we saw the opportunity there, but even more so because we saw the broad graphics interface opportunity.

Q Now you talked a little earlier about Visual C++. Did Microsoft do anything else to assist third-party developers in writing Windows applications?

A Yeah. We created a large set of tools that would help people write Windows applications. We also did a lot of events, what we call developer conferences, where we would talk about what we were doing, show off the tools, sit and

get feedback from developers. We created a group, the developers relationships group, the acronym is DRG, that would go out and talk to ISVs. So we invested a great deal in trying to make it easy and encourage people to write applications for Windows.

Q What was MSDN, which is something that the jury has heard about?

A A lot of the documentation for developers we would pull together in this thing called the Microsoft developer network. So the acronym is Microsoft developer network. And the amount of material in there about writing Windows applications over time, you know, got larger and larger. It was the place that we put -- initially on a CD-ROM, but eventually on a Web site. The information made it quite easy to navigate.

Q You mentioned the Microsoft developer relations group. Why did Microsoft have such a group?

A Well, simply to get feedback on the operating system, you know, what things weren't working, what things would they like to see improved, what new features would they be willing to take advantage of, answer technical questions that came up. And so that was a -- we had good people in DRG that facilitated that communication.

Q Did DRG work with ISVs who were developing applications that competed with Microsoft's own applications?

A Yes.
Q Why did you do that?
A Well, we thought having those applications on Windows, on balance was a good thing for us.
Q Even if that meant that you would lose sales of Microsoft Word and Microsoft Excel?
MR. JOHNSON: Objection, leading.
THE COURT: Overruled.
THE WITNESS: That's correct. We decided that we wanted to work with them and chose to do so.
BY MR. HOLLEY:
Q What was the Microsoft Press -- what is the Microsoft Press?
A That was a group at Microsoft who had actually come out with books mostly about developing for Windows and other Microsoft environments. But they did quite a variety of books.
Q And can you tell us what the Microsoft Systems Journal is?
A Yeah. That was essentially like a magazine, a very hard core magazine about technical issues that if you were really into writing Windows applications, you would definitely want to catch the latest issue of the Microsoft Journal.
THE COURT: On the other hand, if you didn't, you

wouldn't.

THE WITNESS: That is certainly correct.

MR. HOLLEY: It might not be on your coffee table, Your Honor.

THE WITNESS: It would be gobbledegook.

BY MR. HOLLEY:

Q Turning to the late 1980s in word processing software on the DOS, MS DOS, was Microsoft the leader in that area?

A No. In DOS word processing, WordPerfect was substantially the company that sold the most MS DOS word processors. And then there was MultiMate, there was MicroPro. Microsoft's Word product was out there, but would have been somewhere fourth or fifth in terms of relative sales volume.

Q Given that the people writing Microsoft Word for MS DOS worked on the same campus as the people writing MS DOS, why wasn't Microsoft Word more popular?

A I think there are several things that explain that. One is that we picked up a design that anticipated that variable sized fonts and laser printers and all these things would come along. So some of the off-putting user interface of Microsoft Word was because we supported those things, but they weren't that interesting until better printers or graphics interface came along.

Also I would say we just didn't do a very good job.

You know, our overall usability in particular was not as good as a number of other vendors.

Q Did the WordPerfect Corporation create a version of its word processing software for early versions of Windows?

A No. They didn't come out with a word processer for Windows until about 1991. So fairly late in the Windows game.

Q What, if anything, did you personally do to attempt to persuade the WordPerfect Corporation to develop a version of its word processer for Windows?

A Well, we devoted resources both in DRG and executive time to communicate with WordPerfect on a regular basis that, come on, you should do a Windows version. Windows is going to be popular. And, you know, Pete Peterson, I remember at one of the industry conferences, said, hey --

MR. JOHNSON: Objection.

THE COURT: Overruled. It's not for the truth of what he said.

Go ahead. You may answer. You can say what he said.

THE WITNESS: Okay.

THE COURT: It's just the jury can't consider whatever you say he said for the truth of what he said is all.

Go ahead and answer the question.

MR. HOLLEY: Don't think too hard about it.

THE COURT: I have been trying to think about computers, so maybe you ought to think about law.

Go ahead.

THE WITNESS: Pete said that he wanted to wait until everybody demanded it and he wasn't going to get out in front on this one.

In any case, they -- you know, they -- everybody knew that we were out talking to everyone trying to encourage them to do Windows applications.

BY MR. HOLLEY:

Q What difference does it make to you whether WordPerfect Corporation developed a version of its word processor for Windows?

A Well, when somebody was considering whether to switch from using MS DOS or -- yeah, to switch from using MS DOS, if they had been using WordPerfect, then not only would they have to buy a computer that ran Windows and get used to using Windows, but also at the same time they would have to buy another word processor, which might not handle their documents well, we would have different sets of commands, wouldn't print their documents out the same way that their WordPerfect word processor had on MS DOS. So it meant the barriers for switching up to use Windows were just that much higher.

If you could switch and WordPerfect had a version of the word processor that read the same documents, printed in the same way, had the same type of command structure, it would be just an easier switch to move into the graphics interface world and use Microsoft Windows.

Q How, if at all, did the fact that WordPerfect competed with Microsoft Word affect your judgment about whether it was a good idea for WordPerfect to create a version of its product for Windows?

A Well, we encouraged them and went to the effort to try and make it so they could do it, and eventually they did.

Q In terms of creating a new operating system like Windows, is it enough to have one or two applications that run on that operating system?

A No. In order for a new operating system to get any volume at all, it's got to have a lot of applications, I mean literally thousands of applications. And so you just need a ton of excitement about something unique that you're doing with that operating system. Now you can write at lot of applications yourself, but you are going to have to get ISVs as well to see it as a big, big opportunity.

Q Is that true even if the two or three applications you have are enormously popular?

A That's not enough because let's take a company, they want to have some uniformity what software is being used

inside the company in terms of how they secure documents, train people to do things. So they are going to pick very few operating systems, sometimes one.

MR. JOHNSON: Objection as to what they're going to pick.

THE COURT: Overruled. I think he's got to take into account what they are going to do in terms of what he does. Overruled.

Go ahead. You can answer.

THE WITNESS: Companies -- when companies pick operating systems, they decide on very few because the complexity of having too many would mean their support staff or figuring out if the data is secure such that it would just be too difficult. And so when they pick a few operating systems, they need to make sure that even the more obscure applications that they need are there. You know, their engineering department needs some applications, their legal department needs some applications. So the total set of applications that would make an operating system acceptable to them is going to have to have a lot of applications, not just a few popular applications.

BY MR. HOLLEY:

Q I would like to switch topics now and talk about something that the jury has heard about, which are suites. What software company was the first to package different

business applications together in a suite?
A Well, in terms of productivity software, Microsoft was the first to create this office concept where a lot of productivity packages would be both technically integrated and available as a group for a much discounted price.
Q Why did Microsoft do that?
A Well, we thought that if we put together the various office components, word processing, spreadsheet, presentations and other things, and made that available about a third the price than if you bought them individually, A, it would be a very good deal for somebody who wanted to use multiple of those modules, and, B, for somebody who had hesitated and, say, only bought one module, they would then find that now they were able to get more modules and would probably start using them, would probably start using PowerPoint for presentations. If they hadn't known the power of the spreadsheet, then they would get in there and start having the benefit of that in terms of improving their productivity.
Q What impact, if any, did this lower pricing that you just described have on sales of suites?
A Suites definitely caught on. So you had -- you know, you had the move towards Windows and graphical applications, and Macintosh. And then you had this move towards suites happening in the early '90s in a big way. Suites became

quite popular.

Q What, if anything, was the WordPerfect Corporation doing in the suites segment of the software business in the early 1990s?

A Well, everyone who had had a popular application had to decide did they want to participate in the suite business or not. And so they had, you know, several ways to go about that if they did. They could assign an internal engineering team to try to create those other products. They could buy or license other products through some type of alliance and assemble a kind of suite.

Now one of the aspects of a suite is you want to have the ability to move information between the products and you want to have a very common user interface. So having the same set of engineers work on the suites would allow your suite to be a lot more consistent than if they come from different places.

WordPerfect in particular, they, in the DOS environment -- of course, their word processor had been very popular. They had had on some other environments some spreadsheet work, but it wasn't very powerful, not very popular, so they looked at whether they could ally with other people who had spreadsheets. At one time they talked to Lotus, who had the 1-2-3 spreadsheet. Later they talked to Borland, who had the Quattro Pro spreadsheet. So they

went out and tried to find things to assemble into a suite.

Q Now you mentioned moving things from one application in a suite to another. Did Microsoft ever develop any technology along those lines?

A Yes. We created initially in the Office group, but it worked out so well that we put it down into Windows so everybody could use it, the ability to have in one document information from another application.

And so a classic example would be in Microsoft Word, if you want to chart, well, Word doesn't have any ability to make a pie chart at all. But Excel, which was the spreadsheet, had the ability to make that pie chart. So what it let you do was go point to that chart and move it into the word document. And then it would be sitting in that word document, so whenever you would read it, it was there. Or if you wanted to edit it, you would just point to it, click on it, and actually it would realize. So that came from Excel and it would run Excel.

Now doing that so the user interface is fairly clean, it didn't take too much memory, it wasn't too slow, it was very tricky. It had the kind of obscure names -- this capability I'm talking about are object, linking and embedding. They called it that because this ability to put something in another document, that was the object. And then you would embed it in one application, but it was still

linked back to the original application, so when you clicked on it, you could still edit it. So it was called object, linking and embedding. Because everything had an acronym, it was also called O-L-E, OLE.

Q When Microsoft moved OLE from Microsoft Office to Microsoft Windows, how, if at all, were ISVs benefited by that?

A Well, they had the opportunity to use it. Some ISVs chose to use it. Some chose not to use it. It was a feature in Microsoft Office that a lot of users found -- that Microsoft Office used, a lot of users found attractive. So over time more and more ISVs included that capability in their applications.

Q Are you familiar, Mr. Gates, with a product that was called Borland Office?

A Yes.

Q What was Borland Office?

A Borland Office was where Borland, which had the number three spreadsheet called Quattro, entered into an agreement with WordPerfect to combine their Windows word processor together with Quattro Pro and offer that as one of these suite type products.

Q Was Borland a consumer brand at the time Borland Office existed?

A No. Borland was mostly known for their development

tools. They had written a pascal compiler that was fairly popular. So when they came out with a spreadsheet, it was kind of a different area for them to be in. And, you know, they never really got the prominence that 1-2-3, which was the Lotus spreadsheet, or the Excel, which was the Microsoft spreadsheet, got.

Q Did you view Borland Office as a serious competitor to Microsoft Windows -- excuse me, Microsoft Office?

A Yeah, it was a competitor. It was one of the suite competitors. It never achieved a large market share, but we took it seriously.

Q Now I would like to turn -- to move forward a little bit to 1994, and I'm going to ask you to take a look at what's been marked as Defendant's Exhibit 297.

Mr. Gates, can you tell us what Defendant's Exhibit 297 is?

A Yeah. This is an e-mail -- and I don't know how much people in the courtroom have seen this before, but it's an e-mail that I write where it says from Bill Gates, and it's to some Microsoft people, Chris Peters is one of them, Hank Vigil, Peter Pathe, and then copied to five other people with the subject -- because I'm forwarding it -- re: Word won InfoWorld on February 7th, 1994.

Q And can you tell us who the people are that -- not obviously their titles, but if there is any grouping of

these people that you are sending this e-mail to?
A Yeah. These are people who are in the Office group. What happened here is that we had gotten an e-mail telling us that one of the computer magazines, the one whose reviews were the most watched and the most important at the time, a magazine called InfoWorld, had reviewed Windows word processors recently. And so the first e-mail in the chain is from the people telling us that Windows Word won that review.

Then Chris Peters, without adding text, he just sends that to me. And then I take that and send it to a lot of people, but I had that text at the top about what an important development it was. So these are people in the Office group that I'm explaining my view of what a great achievement this is.
Q You say in the first paragraph, Mr. Gates, an important win. This is super. Why did you say that?
A Well, in the DOS word processing business, WordPerfect had two really key things that had driven their success. One had been that they had won the InforWorld reviews again and again and again. So with their DOS word processor, they added features. They did what users wanted. They had had the right speed. So they stayed ahead in the reviews.

The second thing was they had this free support where you could call up an 800 number and talk as long as you

wanted, which was a significant expense for them, but it was something they provided to their users.

So those two were two key things. In fact, all of their ads would simply take their support policies and their InfoWorld review information, and that was the DOS WordPerfect had.

Now here in the Windows world, their Windows word processor was not nearly as good. So if you look down in this e-mail, it says Word 8.2, Ami 7.0, WordPerfect 6.0. So InfoWorld would rank things on a scale of one to ten, and a six was kind of a so-so ranking and an eight or nine was a very, very good ranking.

And so when it says we won, we won that because our score was higher, it says, especially the gap between us and WordPerfect. That's the gap between the 8.2 and the 6.0. Now we knew -- you know, we were working hard on Windows Word. We knew it was a great product. But it, you know, was a great confirmation. Then I go on to say, InfoWorld made WordPerfect and this will help unmake them.

Q What did you mean by that, sir?

A Well, I'm just referring again to the fact that they had touted their winning the InfoWorld reviews with their DOS product as a key part of their positioning. If you had asked people about DOS WordPerfect, what ads they ever ran, it was the free support policy and the product that got the

best reviews. And so I was saying that it was a -- you know, it was a real milestone that, you know, they were no longer doing the word processor that got good reviews.

Q Now in the next paragraph, Mr. Gates, you say, I can't believe the string of news about WordPerfect. First we start with no real suite strategy.

What did you mean by that?

A I meant that they didn't have a set of uniform and strong products to put together into one of these Office suites, that is even though they had done some stuff with Borland, the Quattro Pro user interface was so different, and even the way you got help was different than the way you did it in WordPerfect, so it was kind of incoherent. And they didn't really have a strategy for the other elements, what were they going to do about presentation and database. So they were challenged in terms of coming up with a suite strategy.

Q Then you go on in the sentence, Mr. Gates, to say, then we get a new CEO who adds the following.

What significance, if any, did you attribute to the fact that WordPerfect had a new chief executive officer in 1994?

A Well, when a new CEO comes along, it's always an opportunity for a company. It can be a good time where they pick some new strategy that makes more sense, get rid of the

software that didn't make sense, or it can be just a time of disruption where the CEO picks some things that are not going to work and doesn't really gel with what the actual strengths of the company are.

THE COURT: Mr. Johnson, I could be confused about this. To help the jury, you may want to identify who the new CEO is.

THE WITNESS: Bob Frankenberg, who had recently, not long before this e-mail -- sorry.

MR. HOLLEY: I don't think either Mr. Johnson or I should be testifying.

MR. JOHNSON: I'm not saying a word.

BY MR. HOLLEY:

Q We're now talking about February 1994, Mr. Gates. So do you know whether WordPerfect had a new CEO, not Novell?

THE COURT: Go ahead. That's precisely why I asked the question because I didn't think it was Mr. Frankenberg.

Who did you think it was?

THE WITNESS: That's a good question.

THE COURT: I don't think that Mr. Johnson would object to you leading on this one.

MR. JOHNSON: Not at all, Your Honor.

BY MR. HOLLEY:

Q So Mr. Rietveld had replaced Mr. Peterson?

A That's right.

THE COURT: Just in terms of the time frame, this is February '94 and it's not until later?

THE WITNESS: That's right.

MR. HOLLEY: Thank you, Your Honor, for clarifying that.

THE COURT: Just doing that for the jury.

MR. HOLLEY: Okay.

BY MR. HOLLEY:

Q So back to this, you attributed significance to the fact that Mr. Rietveld had replaced Mr. Peterson, and then you say, laying off 20 percent.

What significance did you attribute to that?

A Well, WordPerfect in the earlier periods had been a going company where they added people and people who worked there had a lot of job security. And, you know, I thought, like Microsoft, I think they created a close relationship with their engineers. When you get into a situation then where you're laying those people off, that idea of, hey, we're all in this together, we're going to work super hard, you don't get that same type of positive spiral.

So having a layoff, not only does it have the direct consequence of reducing your capacity by that percentage, it also kind of creates a negative aura, certainly an aura of who is going to get laid off next. So anyway, it's not a

good situation for a software company and clearly was going to hurt their productivity.

MR. JOHNSON: Your Honor, move to strike that answer. He's no expert in the industry of lay-offs or what that means to a company.

THE COURT: Overruled. He is a chief executive.

Ladies and gentlemen, consider it for whatever you consider it for.

BY MR. HOLLEY:

Q Your Honor -- Mr. Gates, what significance, if any, did you attribute to your second point, getting rid of free support policy?

A We were very surprised that they got rid of their free support policy. Of course, it cost them a lot of money because when people would call in on that 800 number, they were being billed per minute, and it made it very easy for people to call up and they had a high volume of calls. But from a competitive point of view, their decision not to bear that expense took one of the things that they had really stood for and got rid of it.

So it -- you know, we viewed it as likely to cause a lot more people to take a hard look at another word processor.

Q Your third point, what significance, if any, did you attribute to no more upgrade of DOS WordPerfect?

A Well, most of their users by far were using their MS DOS product. And so for somebody who's thinking, hey, this thing is still missing a few features, I will stick with it, maybe there will be an update that I can get that will give me those new features, you know, new printers come along, I will get updates that will support those things. By actually announcing that you don't plan to improve your MS DOS product, it makes people realize, wow, I can't stick with MS DOS if I want new word processing features. I'm going to have to switch over to Windows. As long as I'm switching to Windows, there's a question whether you want to stay with the same word processor, or also switch that at the same time. So they kind of created a dead end for the install base user on their DOS product.

Q I think we've already talked about item D. What significance did you attribute to Microsoft coming up with great ads explaining our support quality, if any?

A Well, I had been very critical of the ads that this group had done in a prior time period. They had improved them enough that I was saying, hey, now you're actually running decent ads -- great ads explaining that we have good support too, so that users wouldn't have the perception that only WordPerfect was offering high support quality.

Q I would like to show you now what's been marked as Defendant's Exhibit 315.

Mr. Gates, can you tell us what Defendant's Exhibit 315 is, please?

A Yeah. This looks like the document I wrote at the time when I heard that Novell was buying WordPerfect and acquiring Quattro Pro from Borland and talked about my reaction and some of the possibilities coming out of this business combination.

Q Now directing your attention, sir, to the first paragraph, you write, the already intensely competitive software business has become even more competitive.

What did you mean by that, sir?

A I meant that if they were able to take the set of products and the scale that this new combination would create, that, you know, they would be a better competitor. They would have more pieces that they could put together. I already viewed the software industry as fast moving, highly competitive, products getting better, prices going down, but this looked like at least it gave them a chance of being a stronger competitor.

Q Now under the heading key impacts under the subheading Office, you wrote if Novell executes well, they will be able to turn their Office suite into a serious contender.

Why did you write that, sir?

A Well, at the time Windows WordPerfect was not achieving high market share and they didn't have a particularly strong

suite offering. But my view was that if they executed well, if they took the resources that would be available and did a very good job, they would have a better suite product. So they had a -- it was a chance with the right execution for essentially a comeback.

Q In the second sentence, Mr. Gates, you write, the fading strength of WordPerfect and Quattro could be reversed.

Could you explain what you meant by that, sir?

A Again, both those products were not achieving a high share. And if you could have gotten all the Quattro Pro engineers, hired them in, got them very quickly to change the user interface to be consistent with the WordPerfect interface. Even the most basic things were quite different between those two products. It was bizarre that they even previous to this had been put into a suite. You could actually take some of those problems, you know, then take the resources of Novell, if you hired more engineers, you could fix some of the big things that InfoWorld had pointed out were the missing features and the clunkiness of the Windows WordPerfect product. You had a chance to both fix the products individually and integrate them in a stronger fashion.

Q Now moving down the page under the subheading business unknowns, I would like -- just because you just talked about

it, I would like to look at the third sentence which says, Quattro Pro developers are being excised from the barbarians world to a completely new company.

What did you mean by that, Mr. Gates?

A Well, you know, in this sort of memo I'm talking about some of the possibilities, and we had seen mergers before and many of them in the software industry were not well executed. You know, so here, under business unknowns, I talk about how people will be diverted for six months, you know, it's hard -- it's hard to do.

Quattro Pro was from Borland and Borland had a very, very different culture than most software companies and, in particular, what I knew of the WordPerfect culture, that is the developers didn't have to come in during the day. They were kind of prima donnas. They even referred to themselves as barbarians, that is they liked that label. The CEO himself of Borland, a guy named Philippe Kahn, liked to think of him as the ultimate barbarian and his developers as the kind of barbarian type who would just come in and work all night and do things.

So, you know, it's going to be interesting to see how those people, if they joined the combination and were asked to do these quality improvements that were needed and would help the company, to see if they made that transition, would they stay, would they work well in that environment or not.

Q Now skipping a sentence and then going to one that begins, Novell's leadership will be stretched incredibly thin by absorbing all of these pieces, particularly with the succession plan unclear.

What did you mean by the succession plan?

A Well, at the time this business combination was announced, they had not yet said -- and I don't know, they may not have known themselves. They didn't say who was actually going to run the thing. So it was kind of -- that was kind of the surprising thing to put very disparate pieces together that had very different code bases, very different user interfaces, put them together and, you know, say they were going to execute as a competitor for Lotus and Microsoft and not say who was going to be in charge of the whole thing.

Q Who had been the CEO of Novell up until this time?

A I think Ray Noorda was still the CEO up through the this time period.

Q Who replaced Mr. Noorda?

A Eventually Bob Frankenberg took the position to run Novell, and including these new assets that were part of the company.

Q Did you have any view on Novell's choice of Mr. Frankenberg?

A Well, he's a good person, but his background was at

Hewlett Packard, which was a hardware company. When they tried to do software, like Open View, it hadn't been successful. So I thought he was an unusual choice, but, you know, not a bad person.

Q Now turning to the second page of this document under the subheading technology unknowns, you write, Novell has never had a technical agenda that its product was designed around.

What did you mean by that, Mr. Gates?

A I meant that they hadn't come up with architecture that really would show how their products would fit together. The original product NetWare, that Drew Major and some other people created, was a great piece of work. It was very efficient, very well tuned for the one thing that it did, which was file sharing. But once they had done that, they never really got a plan for where they wanted to go around. So I was commenting on that they didn't -- they didn't have a technical agenda that rallied their developers to work towards a common set of architectural goals.

Q What significance, if any, did you see that having for their suite product?

A Well, when you do a suite product, where they come -- the suite elements come from different backgrounds, there is a lot of architectural decisions to be made. Here I talk about what are they going to do about the database

component. You know, Microsoft Office had this module called Access that was very, very popular. And here they just had a temporary license to a database product.

They didn't have a way of writing macros, doing automation across the different modules. So I talk about, well, are they going to use the Quattro Pro automation language, which really is very structured specific, or were they going to use what was in WordPerfect that was very limited. You know, they had to make some choices and drive everybody to a common technical agenda and do it fairly rapidly to take what were essentially assets that were losing share and see if they could actually, through reenergizing them, get them back onto a positive track.

Q I think this may be a new term or a new concept for people in the courtroom, at least some of them. Can you explain what an automation language is in the context of a business application?

A Yeah. Even when you work with a spreadsheet or a word processor or a database, you might want to write a little program. So say you want to print mailing labels, the actual format of the mailing labels you might want a little software that decides how you're going to organize those things. Of if you want to do a report that presents information, you might want a little bit of logic that determines what numbers are in that report.

So even at the level of these Office productivity applications, there was the ability to write what were sometimes called automation programs, or macro programs, that take the capabilities of the Office pieces and tell them to do something. And once you put things in a suite, you would want to use one language, one automation language across the different modules so that, A, you only had to learn one, and, B, if you wanted to do a task that involved multiple modules, then that automation link would be able to perform them.

Q Did Microsoft write any automation languages for use with Microsoft Office?

A Yeah, that was a huge strength of the Microsoft Office suite is that we had a way of using visual basic programming to control all the different modules. We had one automation strategy across all the products.

Q What was that product called, the automation language for Office?

A Well, it was sort of Visual Basic for Office, or Visual Basic for Applications. And because everything has an acronym, that was called VBA, Visual Basic for Applications.

Q Was VBA available only to Microsoft applications and developers?

A No. In fact, it was completely optional. Some Windows developers took advantage of the fact that we offered VBA,

actually put that into their productivity software, and some did not. But we got a lot of people to take on VBA, and that became a selling feature for the people who actually chose to do it.

Q I would like to switch topics now and talk about something called middleware. The jury has heard quite a lot about it.

THE COURT: Can we go another ten minutes? Everybody okay? We're switching topics. This would be a good time to take a break. If everybody is okay, we'll go another ten minutes or so.

BY MR. HOLLEY:

Q Mr. Gates, is there a generally accepted definition of the word middleware in the software industry?

A No. It's used in different ways by different people.

Q I'm going to ask you to accept the definition of middleware for the next series of questions and, you know, you can tell me if you disagree with any of these elements. But let's look at demonstrative Exhibit 186.

Now, Mr. Gates, just so we're on the same page in terms of terminology, if I say to you that a piece of middleware is cross-platform, what does that mean to you?

MR. JOHNSON: Your Honor, I would just like to note an objection to this slide. Professor Noll didn't say that. And so we would just have a continuing objection to

the use of that slide.

THE COURT: Nobody -- this is Mr. Gates' understanding.

Go ahead.

MR. JOHNSON: Your Honor, it says that this is -- the source of this list is the testimony of Roger Noll on the bottom, and we would just raise an objection, that's not what he said.

THE COURT: If it is, you all can argue that to the jury, whether or not that's -- I'm sorry. I was -- I didn't see that.

Ladies and gentlemen, forget the source. That can be argued to you, what the source is. This is Mr. Gates' understanding. Whether it's the source or not, you ought to determine later.

MR. HOLLEY: Thank you, Your Honor.

BY MR. HOLLEY:

Q Mr. Gates, back to my question, if I say to you that a piece of middleware is cross-platform, what does that mean to you, sir?

A That means it's available to run on a number of operating systems. So, you know, generally the popular -- it would have to be available on popular operating systems.

Q And what does it mean to you to say that a piece of middleware is ubiquitous?

A Actually the parenthetical makes that clear, that is it has to be out on a huge number, a very high percentage, you know, say over 80 percent of all the machines out there, so that a developer would consider requiring it for their application to run on.

Q And the third element here on this chart, what does it mean to you to say that a piece of middleware exposes enough APIs to allow ISVs profitably to write full-featured personal productivity applications that rely solely on those APIs?

A Well, it means that if this -- if the middleware -- to meet this definition, it has to have enough power that literally you write -- something like a word processor, so that just by the middleware being available, the word processor runs on any platform where you have that middleware and runs in a decent way, that is people really want to use it, it's fast enough, it fits in with other things they do on that platform. And the way to get there is you would have to have a lot of capabilities exposed through application programming interfaces, which is that acronym API, to allow the software developers, that's the ISVs, to say that yes, they wanted to go ahead and create such applications.

Q As we're here in November 2011, has any middleware ever been created that meets those three conditions?

A No, sir.
Q Has anyone ever tried to do that?
A Well, you could say that some of the stuff that the Java -- Java script browser people talked about doing, they talked about aiming to get to this point. Now they never actually got there. There isn't, you know, a single productivity application that ever relied just on those APIs. But, you know, in their prognostications about where they wanted to go with it, it was something they talked about wanting to achieve, but they did not.
Q Does Microsoft Office expose any APIs?
A Yes, it exposes APIs.
Q Does it meet these three conditions, Microsoft Office?
A No. You would never -- the API set exposed by Office is quite narrow. It's specifically for people who want to do enhancements on top of those Office applications, which would be fairly narrow vertical type applications. So very different than the API that's down in Windows itself.
Q Now you used a term I don't think we've heard before, which is vertical application. What do you mean when you say vertical application?
A I mean having a very targeted purpose. So, for example, if there is a way that, say, a baseball team likes to print out summaries of games, there is a little piece of software that runs on Microsoft Word that makes it easy to

print out box score type things, statistic type things. So generally these little applications are for quite a specific purpose and they tend not -- you know, compared to a full-blown application, they would have like a few percent of the lines of code, that is they would be hundreds of lines of code instead of tens of thousands of lines of code. So very different. Nobody would ever be confused about Word as a platform versus Windows as a platform.

Q I wish I could share your optimism on that point, Mr. Gates.

THE COURT: I will strike that one.

BY MR. HOLLEY:

Q Mr. Gates, were you ever concerned that a product called WordPerfect was middleware under this definition?

A No.

Q Why not?

A Because WordPerfect, what it exposed was very similar to what Word exposed, that is the type of extensibility that somebody who would program on top of a word processor would need. So none of the things that you associated with an operating system or that you would need for productivity applications, like deep memory management, device management, task management, none of those things are in productivity application APIs, neither WordPerfect or Word or other productivity applications. So you would never

think of them as middleware.
Q What if I added two other software products to WordPerfect, if I asked you WordPerfect plus something called AppWare plus something called OpenDoc, did you ever regard that combination as middleware under this definition?
A No.
Q Why not, sir?
A Well, in the case of OpenDoc, it was one of those technologies where it never actually became concrete. Apple had done some work and a bunch of other people got together and said, okay, let's do some non-Microsoft sort of OLE type thing. So they talked about did they want to do a common format, did they want to do a common API, and the whole thing fell apart before it became a concrete thing because the disparate platforms that were involved in the particular technical choices that had to be made just weren't going to be satisfactory to all these -- the people who got together.
In the case of AppWare, I don't know that any applications were ever written to that. It was something that I think Novell bought, but I don't know why they bought it because they never -- they never convinced people to write applications for it.
Q Are you familiar with a product called PerfectOffice?
A Yes.
Q What was PerfectOffice?

A Well, sometimes WordPerfect, when they put together their horizontal productivity applications in a group, that was the marketing term they used.
Q At any time did you ever regard PerfectOffice as a middleware threat to Windows under this three-part definition?
A No.
Q Why not?
A Well, it didn't meet any of the three criteria. It didn't expose enough APIs to allow ISVs to write a productivity application. Nobody would seriously contemplate doing that with PerfectOffice.
Q Why? Why wouldn't they contemplate doing that?
A Because --

MR. JOHNSON: Your Honor, Mr. Gates may be an industry expert, but he's not qualified as a technical expert.

THE COURT: Just rephrase the question.

BY MR. HOLLEY:
Q Mr. Gates, based on your experience in competing with WordPerfect in the suite business, did you ever view PerfectOffice as a competitor to Windows?
A No.
Q Why not?
A Because it didn't offer -- it's hard to keep a straight

face. It didn't offer a set of APIs and applications that anybody would ever say to themselves, hum, should I buy Windows or should I buy PerfectOffice. That never would come up.

MR. HOLLEY: Your Honor, I'm about to change topics.

THE COURT: If this is a good time. I'm ready as soon as everybody is. About ten minutes.

MR. HOLLEY: Thank you, Your Honor.

(Jury excused)

(Recess)