UNITED STATES DISTRICT COURT
FOR THE DISTRICT OF COLUMBIA

STATE OF NEW YORK, et al	.	
	.	CA No. 98-1233(CKK)
Plaintiff,	.	
	.	Washington, D.C.
v.	.	April 10, 2002
	.	2:04 p.m.
MICROSOFT CORPORATION,	.	
	.	
Defendant.	.	Volume 15

. . . . . .

AFTERNOON SESSION
TRANSCRIPT OF TRIAL RECORD
BEFORE THE HONORABLE COLLEEN KOLLAR-KOTELLY
UNITED STATES DISTRICT JUDGE

APPEARANCES:

For the Plaintiff:
Litigating States
WILLIAMS & CONNOLLY
Brendan K. Sullivan, Esq.
Steven R. Kuney, Esq.
Kevin Hodges, Esq.
725 Twelfth Street, NW
Washington, DC 20005
202.434.5800

For the Defendant:
SULLIVAN & CROMWELL
John L. Warden, Esq.
Steven L. Holley, Esq.
Richard J. Urowsky, Esq.
Ronald J. Colombo, Esq.
125 Broad Street
New York, NY 10004
212.558.4000

For the Defendant:
WINSTON & STRAWN
Dan K. Webb, Esq.
35 West Wacker Drive
Chicago, IL 60601
312.558.5600

Scott L. Wallace, RDR, CRR
Official Court Reporter

APPEARANCES (Cont.)

MICROSOFT CORPORATION
William H. Neukom, Esq.
David A. Heiner, Jr., Esq.
Thomas W. Burt, Esq.
One Microsoft Way
Redmond, WA 98052-6399
425.936.8080

ALSO PRESENT:

LEVINE, SULLIVAN & KOCH
Jay Ward Brown, Esq.
1050 Seventeenth Street, NW
Suite 800
Washington, DC 20036
202.508.1125

Court Reporter:

SCOTT L. WALLACE, RDR, CRR
Official Court Reporter
Room 6814, U.S. Courthouse
Washington, D.C. 20001

Proceedings reported by machine shorthand, transcript produced by computer-aided transcription.

Scott L. Wallace, RDR, CRR
Official Court Reporter

P R O C E E D I N G S

THE COURT: All right, good afternoon, everyone.

THE COURTROOM: Good afternoon, Your Honor.

THE COURT: We're proceeding with Dr. Appel and Mr. Holley continuing with cross.

MR. HOLLEY: Yes, Your Honor.

CONTINUED CROSS-EXAMINATION OF ANDREW APPEL

BY MR. HOLLEY:

Q. Dr. Appel, could you turn, sir, to paragraph 26 of your written testimony which appears on page 10.

A. Yes.

Q. Now, you say there in the first sentence, sir, that "an operating system is software that manages and controls a computer's hardware and provides a platform on which application programs or middleware can run."

And do you continue to agree with that assertion, sir?

A. Yes, an operating system does that.

Q. And if Microsoft's obligation under Section 1 of the non-Settling States' proposed remedy is to ensure that after Microsoft middleware products are removed, the operating system continues to perform effectively and without degradation, how can it continue to serve the function of a platform for applications if things that applications rely on are no longer there?

A. It's my recollection that there's a parenthetical that says, "except for the functionality that has been removed." Precisely it says, "without degradation, other than the elimination of the functionalities of any removed Microsoft middleware products."

Q. But to the extent that you believe that the purpose of an operating system, one purpose of an operating system is to serve as a platform for applications, you would agree with me, would you not, sir, that that function is impaired to the extent that Microsoft middleware products that are relied on by some of the 70,000 Windows applications are removed from the system and no longer available?

A. Yes, in the sense that some of the functionality will be gone.

Q. Now, you believe that under Section 1, after it comes into effect, OEMs like Compaq and Dell can choose which Microsoft middleware products they want to remove from the operating system; is that right?

A. That's right.

Q. And you also believe that third-party software developers might decide that whereas their applications today run on every brand of personal computer that's running Windows XP, in the future under Section 1, those applications might only run, for example, on the Compaq version of Windows XP because that is the only version of

the operating system that exposes all of the functionality that the software application developer -- that the software developer's application needs to run?
A. You're saying that if Compaq is the only OEM not to remove a Microsoft middleware, and that the other OEMs who remove that Microsoft middleware don't put in some non-Microsoft middleware substitute so that only the Compaq version of the operating system has this particular API support in it?
Q. Yes, that's my hypothetical.
A. Then there could be some applications that would run only on the Compaq configuration of the unbound operating system.
Q. And as a result, some software developers might decide that whereas now they get all PCs running Windows XP, in the future created by Section 1, they might target only Compaq PCs; is that right?
A. Well, what they could do to make their software run on other PCs is to ship Microsoft middleware with their applications, which of course they would need Microsoft's permission to do under license.
Q. And if they did so, their products would become both -- both more complex and larger, as a result?
A. Well, I know when I purchased Microsoft Office, it ships with a copy of the Internet Explorer middleware, just

in case the version of Internet Explorer on whatever version of the Microsoft operating system I have previously installed is the wrong one. That is, there has been many versions of Microsoft Internet Explorer over the years. Microsoft would like Office to run and it uses Internet Explorer, the current version of Internet Explorer, as part of its platform. So when I buy Office for my PC, and the Office I buy today for my PC might run on the Windows 98 operating system or the Windows XP operating system, and Microsoft isn't sure which version of Explorer I have installed, so Microsoft packages Internet Explorer middleware on the same disk with its application and middleware programs in Microsoft Office.

I'm buying Office, but Microsoft has packaged Explorer with it just in case I don't have the right version of Explorer on my computer. And if I do have the right version, then the packaged copy of Explorer won't bother to install itself.

Q. Can you answer the question I asked you, Dr. Appel, which is: Would that make those third-party developers' products larger and more complex, yes or no?

A. It would mean that distribution in the case of distribution on a CD-Rom would be bigger, and the distribution in the case of over-the-network downloading would be bigger, in the case that a copy of the middleware

also had to be bundled with it.
Q. You do not know, do you, Dr. Appel, whether Windows components that fall within the definition of Microsoft middleware products in the non-Settling States' proposed remedy that need to be optionally removable under the definition x(i) may nonetheless be removed by OEMs and third-party licensees in another manner than the manner provided by Microsoft?
A. Well, I do know that -- are you asking me whether there are technical means of removing the middlewares from the operating system product?
Q. Well, don't you believe that one possible interpretation of the non-Settling States' proposed remedy is that Section 1 requires Microsoft to provide a technical mechanism for OEMs and third-party licensees to remove components that fall within the definition of Microsoft middleware products and that Section 2.c. little Roman iv permits those OEMs and third-party licensees to remove components from Windows by technical means other than those that Microsoft has provided under Section 1?
A. Well, first of all, Section 1 requires that the Microsoft middleware products may be readily removed, and by "readily" I take that to mean whatever technical means are available to the OEMs must not be unduly difficult, okay.

In Section 2, 2.c.(iv), at the very end of the paragraph where it says that "Microsoft may not prohibit," I guess by license, the OEMs from removing the code from Microsoft middleware products, I interpret that to mean that the OEM could remove a Microsoft middleware product, not just any arbitrary piece of Microsoft middleware, but a Microsoft middleware product from the operating system.

Q. Well, take a look, if you would, sir, at your deposition, the second volume of page 289 starting at line 17, and tell me when you're there.

A. What page did you say?

Q. 289.

A. Yes.

Q. Now, do you recall being asked the question: "And if there is, does Section 2.c. little Roman iv give OEMs and third-party licensees the right to remove those Microsoft middleware products even though they are outside the scope of the unbound version?

"Answer: I don't know. It's possible that the answer is yes, and in that interpretation, Provision 1 requires that in the first iteration that Microsoft provide certain technical means that OEMs can use to remove Microsoft middleware products. And one interpretation of Section 2.c. little Roman iv might be that OEMs are permitted to remove the code for Microsoft middleware

products by technical means other than what Microsoft has provided by Provision 1. But I'm not sure."

Do you remember being asked that question, sir, on March 13th, and giving that answer?

A. Yes, I do.

Q. You do not know, do you, sir, how many cross-dependencies there are between the component, the components in Windows XP Embedded that you associate with Internet Explorer and other parts of the operating system?

A. No, I don't. I thought about how one might measure that, but it's not something that I've been able to do in the last two or three weeks.

Q. Do you believe, Professor Appel, that a principle of modular programming, one principle of modular programming is that the interfaces exposed by a block of software code should be as small as possible relative to the implementation of functionality that lies behind those interfaces?

A. Yes.

Q. And one benefit of having small interfaces is that they permit software developers to alter the way in which the functionality exposed by those interfaces is implemented within a block of software code without affecting the external interfaces of the code, correct?

A. Yes, that's one reason, for example, why the States'

remedy limits itself to only a certain set of APIs that need to be exposed and doesn't try to interfere with Microsoft's discretion with arranging its internal APIs.

Q. And even if blocks of software code are designed in accordance with this principle of modular programming, changing software code within one module can have unforeseen effects in other modules?

A. Yes, that's true.

Q. If a third party like Novell has a block of software code that it wants to use as a substitute for something that falls within the definition of a Microsoft middleware product, and the lines that define that block of software code do not correspond with the modules within the Windows operating system, is Microsoft required under Section 1 to recraft its code so that the Novell block fits precisely into Windows?

A. No. The -- it's not the case that any arbitrary fragment of Microsoft middleware must be removable under Remedy Provision 1; it's that an entire Microsoft middleware product may be removable. The boundaries of what is an entire Microsoft middleware product is not really at the discretion of Novell.

Q. So, Microsoft gets to decide what constitutes a Microsoft middleware product? And I know this is a simplification, but let's -- tell me if you can't accept

it, but if the block of software code that Microsoft makes optionally removable is square, and the Novell replacement is hexagonal, such that it won't fit in the operating system, that's not Microsoft's problem under Section 1 as you understand it?

A. Yeah, let me rephrase your question. If Microsoft makes blocks of middleware code removable at the boundaries of the Microsoft middleware products in compliance with this judgment -- and the definition of Microsoft middleware product gives some guidance about what those boundaries are -- then if Novell wants to fit in a block of code that doesn't precisely match those boundaries in such a way that it won't fit, then that's not Microsoft's problem.

Q. Now, you're familiar with Professor Bennett at the University of Colorado's example in his expert report in this case of a five-function calculator that uses the same shared software code to perform five different functions, addition, subtraction, multiplication, division, and square roots, correct?

A. Yes, I read his expert report.

Q. And you believe that that example is reasonable as a matter of software engineering, do you not?

A. As a way to build a program, a calculator, yes.

Q. And you also agree that if you removed some of the software code from this calculator example that was relied

upon to support a particular function and you didn't replace it with a functional equivalent, then other functions of the calculator which also relied on that same shared code would be disabled?

A. That's right.

Q. I take it from your testimony yesterday that you have formed a conclusion based on your review that you've been able to do to date of Windows XP Embedded that Microsoft's operating system is modular?

A. Yes, I would say it's built in a modular way.

Q. And that modularity does not preclude the existence of cross-dependencies among modules such that if I pull one module out, other parts of the operating system malfunction?

A. Yes, it's normal in modular programming that one modular software program may rely upon another module for functionality. And so if you remove one module upon which another module has relied, then that other module won't be able to obtain that functionality.

Q. And although there is not a precise mathematical relationship, you agree that as a general proposition, the more modules there are in a complex product like Windows, the more likely it is that there will be cross-dependencies among those modules such that pulling one module out will cause other modules in the operating system to malfunction?

A. I'm not sure I would put it that way, that -- as a matter of software engineering, I encourage my students to divide a large piece of software into many small modules, and then to take those small modules and group them into bigger modules and so on, that having more modules is not necessarily something to be avoided.

Q. No, I wasn't suggesting that, sir, but take a look at your deposition at page 173 in the first volume starting at line 14.

Do you remember being asked: "And in fact, the greater the number of modules, the more likely it is that there will be such cross-dependencies, correct?

"Answer: I would hate to make a quantitative judgment of that form, but yes, with more modules there is at least more potential for dependency between modules."

Do you remember being asked that question and giving that answer, sir?

A. Yes, I do. And yes, it's true, with more modules there is at least more potential for dependencies between modules.

Q. Now, turn with me, if you would, sir, in the non-Settling States' proposed remedy to the definition of middleware which appears in paragraph 22.x.(i) and that -- I'm sorry, Microsoft Middleware Product, 22.x.(i) on page 23, and tell me when you're there, sir.

So, this is not a list of particular pieces of code in the Windows operating systems; it is instead a list of categories; is that correct?
A. Yes, it does appear to be a list of categories, so I would imagine, for example, that an e-mail client software product would be a Microsoft middleware product.
Q. And as to Windows XP Professional and Windows XP Home, there are at least two things that would fall within the category of Internet browsers, correct, both Internet Explorer and MSN Explorer?
A. All right.
Q. Well, do you agree with that? I don't want to --
A. I think so. I'm not exactly sure what the difference between Internet Explorer and MSN Explorer is.
Q. Okay. How many different Microsoft middleware products in Windows XP Professional fall within the category media creation, delivery and playback software.
A. I'm not sure. There is the Microsoft -- the Windows Media Player, which I believe falls in the category of delivery and playback software. I'm not sure what products Microsoft sells in media creation. I guess -- I'm not really an expert on the different product categories of Microsoft software.
Q. There are two features of Windows for people who have visual disabilities, one called Windows Magnifier and the

other one called Windows Narrator. Are they in your view media creation delivery and playback software as that category is described in Paragraph 22.x.(i)?
A. I'm not sure.
Q. Let's look at the category of software in 22.x.(i) entitled "Management Software" -- "Directory Services, and Management Software." Is the Disk Cleanup Wizard in the accessories folder of Windows XP, management software as that term is used in 22.x.(i)?
A. Direct -- you mean under "Directory Services and Management Software"?
Q. "Directory Services and Management Software," yes.
A. No, I don't believe it is.
Q. Okay. And what do you think is encompassed in Windows XP Professional within the category "Directory Services and Management Software"?
A. Directory services and management software is software that manages directories in the sense of a certain special kind of database that attributes -- that attributes names of people to their roles in an organization, names of people to which kinds of access privileges they have to different parts of the network, names of machines to, you know, to which people they belong to and where they sit and how they relate to each other.
I believe that Microsoft has a product called

Active Directory that does that, but I'm not familiar in great detail with directory services in general or with Microsoft's products in that area.
Q. Is the Disk Cleanup Wizard in Windows XP Professional systems and enterprise management software as that term is used in paragraph 22.x.(i)?
A. No, I don't think so. I don't think systems specifically refers to computer systems. Enterprise management software, enterprise is, for example, a corporation or a non-profit organization, and so, I believe the software is related to that kind of interoperation between the members of an enterprise. The disk cleanup is related to a specific piece of hardware and so on.
Q. The term "directories" in 22.x.(i) is an imprecise term in computer science which could include a large number of different Windows components, correct?
A. It's my understanding that the use of the term "directories" in 22.x.(i) is consistent and largely overlapping with the term "directory services and management software."
Q. And that is an understanding that you developed as the States' technical expert by calling Carl Ledbetter of Novell and asking him what the term meant in this decree; is that correct?
A. Yes, I did have a discussion with him among other

people in bettering my understanding of directory services and management software.

Q. And that is because the word "directory" has many different meanings in computer science, and you have no expertise in the field of industrial computer science practice that would be sufficient for you to feel comfortable providing a list of what is and is not meant by the word "directories" under Section 22.x.(i)?

A. I am aware of different definitions of the word "directories" in computer science. And the use of directories specifically related to directory services and management software is one that I don't have an extremely great depth of technical expertise in. There are many different kinds of middleware, and I am more expert about some kinds than about others.

Q. Well, you agreed with me at your deposition that the Windows registry in Windows XP could be a directory under 22.x.(i), didn't you?

A. It may have some functionality in common with what directory services and management software does.

Q. And if we got the proverbial computer scientist from Mars to come to the courtroom, he would tell us that a directory in computer science refers to any list of files in a folder, right?

A. That's one of the meanings, list of files,

approximately. That's one of the meanings of the word "directory" in computer science. It's not the meaning of the word.
Q. Right. And the reason that you asked the lawyers at Williams & Connolly and Dr. Ledbetter of Novell what directory meant in 22.x.(i) is from reading this definition, you had no idea, right?
A. I wasn't sure which of the different meanings of the word "directories" in computer science was meant here.
Q. Now, one of the things that's listed as middleware in Paragraph 22.w. on page 22, so it's the previous page to the one we were just looking at, is a network operating system. Do you see that, sir? Actually the definition begins on 22 and the words "network operating systems" appear on 23.
A. Yes, I see that.
Q. And it isn't entirely clear to you what that means because you don't expect to port one operating system to run on another operating system, correct?
A. Yes, I think that network operating systems don't support applications and make them more portable by providing APIs. They make applications more portable by providing communications interfaces. To the extent -- and so most of the kinds of middleware -- many of the kinds of middleware listed here do provide the function of

middleware in making applications easier to port by providing APIs, and so I had been thinking in that mode.

But some of these kinds of middleware render applications more portable by providing communications interfaces. To the extent that the application gets some of its services across the network through communications interface, that means it doesn't have to get those services from the operating system on the desktop machine, and that means that it's easier to port that application to a different operating system on the desktop machine.

Q. You think, Professor Appel, that it would be useful to have a definition of the term "network operating system" in paragraph 22.w. because it isn't entirely clear to you what that term means in this context; isn't that correct?

A. I think I have an understanding of that term.

Q. Well, take a look at your deposition, sir, on page 95.

THE COURT: First or second one?

MR. HOLLEY: I'm sorry, Your Honor, in the first -- they are actually sequentially paginated, and 95 appears in the first of the two volumes.

BY MR. HOLLEY:

Q. Well, actually, why don't you look first, Professor Appel, at page 94 starting at line 9 where I asked you: "Well, where is the term 'network operating system' defined in the States' proposed final judgment?"

And your answer was: "It's not."
A. Yes.
Q. And then -- are you there with me, sir?
A. I'm there.
Q. And then I asked you starting on line 13 of the next page: "Is that true in the case of Solaris, for example, there is a distinction between the version of Solaris that is used to manage a domain and the version of Solaris that runs on any given server in the domain?"
And you answered: "Each machine in a domain probably runs the same version of Solaris, but I think we're referring here to the network management functionality, and it could well be that it would be useful to have an explicit definition of this term, 'network operating system,' in the remedy."
Do you remember giving that testimony, sir?
A. Yes, I do.
Q. Now, Microsoft Office is a suite of business productivity applications, correct?
A. Yes, although it also serves as middleware for other applications.
Q. And you are aware that the Court of Appeals in this case did not hold that Microsoft Office was middleware?
A. I'm not sure of that. And it also may be the case that in the year 2001 or 2002, that Office is serving more and

more as a middleware platform for other applications.
Q. Well, Professor Appel, take a look, sir, if you will, at page 124 in volume 1 of your deposition starting at line 19.
Do you remember being asked, sir: "Did the Court of Appeals say that Microsoft Office was middleware as it used that term?
"Answer: I don't believe the Court ruled that Microsoft Office is middleware."
Do you remember being asked that question and giving that answer?
A. I think I did give that answer, yes.
Q. You do not know, do you, sir, whether the version of Microsoft Office for the Macintosh exposes the same APIs to software developers as are exposed by Microsoft Office for Windows?
A. That's right.
Q. And you believe that if Microsoft Office for the Macintosh does not expose APIs to software developers, then it -- it would not be middleware as the Court of Appeals in this circuit understands that term?
A. That's right. If any particular version of Microsoft Office does not expose any APIs as a platform for software developers, then it's not middleware.
Q. And you believe that any Microsoft application that

uses something called Visual Basic for applications to permit its functionality to be accessed by other software products would be a Microsoft middleware product under the non-Settling States' definition?

A. Yes, Visual Basic is one way of programming applications, and if a Microsoft software product provides APIs that those applications can use as a platform for getting services, then it is middleware.

Q. And you do not know, sir, how many of the hundreds of software products marketed by the Microsoft Corporation would thereby be converted into Microsoft middleware products under the non-Settling States' decree?

A. No, I don't.

Q. I'm sorry, was that a no?

A. What was the question? How many of the Microsoft --

Q. Yes. My question was: Do you know as you sit here today how many of the Microsoft software products would be converted into Microsoft middleware products by virtue of the fact that their functionality is exposed to software developers through the use of Visual Basic for applications?

A. "Converted" is a funny term. I don't know how many should be considered as middleware because they expose APIs for Visual Basic programming.

Q. I'm happy to accept that amendment. And the answer is

you don't know how many, sir?
A. I don't know.
Q. Now, let's turn to Section 4 of the non-Settling States' proposed remedy. You believe, do you not, Professor Appel, that one purpose of Section 4.A. is to permit other companies to create functional substitutes for Microsoft platform software?
A. Yes, that's right.
Q. And that Microsoft platform software would include Microsoft Office, correct?
A. Yes.
Q. And it would also include all Windows operating systems from Windows CE through all of the desktop versions of Windows up through Windows NT 4.0 Server, Windows 2000 Server, Windows 2000 Advanced Server, and Windows 2000 Datacenter Server; is that correct?
A. Assuming that all of those fall under the definition of Microsoft platform software, which I believe is the case.
Q. Now, did the Court of Appeals in this case hold that Microsoft has monopoly power in server operating systems?
A. I'm not sure.
Q. And did the Court of Appeals in this case hold that Microsoft has monopoly power in operating system for non-PC devices?
MR. HODGES: Objection to the extent he's being

asked to testify about what the Court of Appeals held.

MR. HOLLEY: Your Honor, I'm just asking for his understanding having read the opinions, whether he thought that the Court of Appeals held that Microsoft had monopoly power in operating systems like Windows CE.

THE COURT: It does seem to me that for somebody who's an expert, he can indicate if that's his understanding or not. That's the basis that informs part of his decision. He's already indicated that, you know, he's knowledgeable of the Court of Appeals opinion, and he's answered earlier questions.

If he can't, then fine, he'll say so, but I think as an expert, if he's reviewed it, he can indicate whether this is, in his view -- I don't have to be bound by it -- but in his view if it fits into what the Court of Appeals has stated or not, so I'll allow it.

BY MR. HOLLEY:

Q. Professor Appel, do you have the question in mind or --

A. Yes, I believe the Court of Appeals did not hold that Microsoft has a monopoly in hand-held devices -- in operating systems for hand-held devices.

Q. You believe, do you not, sir, that Section 4.A. would require Microsoft to provide competitors like the IBM Corporation and Sun Microsystems with the information that they need to create functional equivalents to all of

Microsoft's operating systems?
A. Well, some of the information that they need: The information about how to interoperate with those same applications that now or in the future might interoperate with Microsoft Windows.
Q. Well, take a look, if you would, sir, at page 130 of your deposition transcript which appears in the first volume starting at line 25:
A. Page 130?
Q. 130, and I think just for context, it might be easier to start at line 13. Do you remember being asked the question: "And what other purpose do you have in mind that you would like" --
A. I'm sorry, what page?
Q. I'm sorry, 130, line 13.
A. Okay.
Q. You were asked the question: "And what other purpose do you have in mind that you would like to be covered by the disclosure requirements?
"Answer: The purpose of providing a functional substitute for Microsoft products.
"Question: And which Microsoft products are encompassed by the notion that ISVs should be able to create functional substitutes under the States' proposed final judgment?

"Answer: Microsoft platform software generally, which includes the Microsoft Windows operating system product and Microsoft middleware products.

"Question: Does that extend to permitting third parties to create functional replacements for Windows 2000 Server?

"Answer: Yes."

Do you recall being asked those questions and giving those answers?

A. Yes, and in the last case I said yes because the Windows 2000 Server operating system, I believe, is substantially the same operating system as the Windows 2000 desktop operating system.

Q. Now, you believe that the functional replacements that IBM and Novell and Sun should be able to create for Microsoft operating systems should be such exact replicas of Microsoft's products that they are capable of substituting for Microsoft's products in existing computer networks such that no changes need to be made when that substitution occurs?

A. Well, I don't think I would use the term "replica," because that carries the connotation of just copying Microsoft's source code, for example. And I certainly don't believe that they should be able to do that.

They need to know what are the functional

specifications of interoperation, how it is that these applications want to talk to the platform software so that they can talk to the applications in the same way. But when the application talks to them in that way and says, "Do this for me," they have to figure out on their own how to do that.

Q. Well -- I'm sorry, I didn't mean to cut you off.

A. Go ahead.

Q. Look at your deposition, page 140. It again begins on line 25, first volume.

"Question: Does it mean that the information disclosures have to be sufficiently broad to create plug replacements for Microsoft products?

"Answer: The disclosures regarding interfaces and communications protocols do have to be broad enough for that."

Do you remember being asked that question and giving that answer, sir?

A. Yes.

Q. Now, one of the things that Section 4.A. is intended to permit Microsoft's competitors to do is to create an alternative to Windows for running 32-bit Windows applications?

A. Yes.

Q. And Section 4.A. would require Microsoft to provide a

precise specification of what functionality is provided to third-party applications by each and every API exposed by Windows operating systems, correct?
A. Exposed by Windows operating systems in such a way that Microsoft middleware or Microsoft applications use them for that interoperation.
Q. And that applies to each and every one of the APIs exposed by Windows, correct?
A. Yes, each API that's exposed by Windows and is actually used by a Microsoft application or Microsoft middleware product.
Q. And even if Microsoft already documents the APIs exposed by Windows sufficiently to allow those APIs to be called upon by third-party applications, Section 4 requires Microsoft to go further than that and to provide sufficient information to permit other companies to replicate the functional aspects of the Microsoft operating system?
A. Yeah. In some cases, more documentation may need to be provided so that the -- so that other developers can speak the same language as the interfaces, yes, in the APIs and communication interfaces.
Q. So even if we -- if I'm correct, as we're standing here today, there is sufficient information available in the world to permit the authors of those 70,000 Windows applications that Judge Jackson found to have written them,

that disclosure is not sufficient under Section 1 of the non-Settling States' proposed remedy unless it also permits other companies to replicate the functional aspects of Microsoft operating systems?

A. Well, again, I'm not sure I would use the word "replicate."

Q. Well, you have, haven't you, sir? Have you used that word in the past in regard to Section 1?

A. I'm not sure.

Q. Well, take a look at your deposition, page 71.

A. Yes.

Q. I said, at line 12: "Question: What, if anything, prevents either Ximian --" and Ximian is the company that is seeking to create an open source version of the .NET framework, is that right, just for context?

A. That's right.

Q. "What if anything prevents either Ximian or anyone else from writing their own data access code to run on top of the common language infrastructure?

"Answer: If the APIs are not fully and clearly documented, or if they're only documented from the point of view of the client of these APIs --" and by that you mean someone calling them to get functionality, right?

A. That's right.

Q. "-- then such implementers may face the same kinds of

problems that implementers have faced in trying to replicate the functional aspects of the Microsoft operating system itself. The APIs are inadequately documented for --" it says "for," but perhaps it meant "from", "--documented for the point of view of providing that functionality."

Do you remember giving that answer, sir?

A. Yes.

Q. And what you're talking about here is disclosures that are sufficient to permit Microsoft's competitors to, in your words, replicate Microsoft's products?

A. Right, to provide the same kind of functionality. The disclosure should say what functionality is provided by the platform software; the disclosures do not need to explain how Microsoft achieved that functionality.

Q. If Microsoft did something innovative in the way its operating system provides services to applications running on top of Windows, the disclosure obligation of Section 4.A. would require Microsoft to hand those innovations over to its competitors on a royalty-free basis under your view, correct?

A. If the innovations had to do directly with the interface, the connection between the application and the operating system, then that would be necessary. That means that a non-Microsoft platform software would not be able to

interoperate at all with the applications for Microsoft's platform software.

If the innovations occurred inside a software module in a way not directly connected with the interface, and the majority of the software code in any large system is in the internals, such innovations don't have to be disclosed under the provisions of the States' remedy.

Q. Now, we talked a little bit earlier today about which interfaces have to be disclosed under Section 4.A. of the States' remedy. You believe that the only interfaces that are immune from this disclosure obligation are those that are naturally completely internal to the operating system kernel, correct?

A. No, I don't think that's true. I gave that as one class of such interfaces that are naturally immune.

Q. Well, take a look, if you would, sir, at page 81 of your deposition, the first volume, starting at line 7, and tell me when you're there, please.

Do you remember being asked the question: "Okay, in how granular a way does the States' proposed final judgment seek to permit people to write replacements for the operating system block numbered 6?" And this is a reference to a diagram in your expert witness report, which you recall, correct?

A. Yes.

Q. And the answer that you gave was: "That at the granularity, let's say, of the entire operating system kernel, that interfaces that are naturally completely internal to an operating system kernel need not be exposed, so that replacements need not be enabled at a granularity layer smaller than the operating system kernel."

By which you meant to say that everything outside the kernel would have to be replaceable, correct, sir?

A. No. What I meant to say is that everything inside the kernel would not have to be replaceable. What I said was that interfaces internal to the operating system kernel need not be exposed. That's not at all the same thing as saying other interfaces all need to be exposed.

Q. Now, when you used the word "kernel" here in your deposition answer, in light of our conversation this morning, would you now choose to amend this answer to say that it is interfaces that are naturally completely internal to the core operating system?

A. If an interface is internal to the core operating system in the sense that it's not called upon by Microsoft middleware products or by applications, then it need not be disclosed under the terms of the States' remedy. That's what I would mean by internal to the Windows core operating system.

Q. Well, we have the problem that we talked about before

lunch, right, where we have to -- Microsoft would have to disclose all of the APIs that are relied upon by each Microsoft application to interoperate with Microsoft platform software, and under one plausible interpretation of that, we're talking about the interfaces between anything that might be viewed as an application level program within Windows?

A. I think before the break I explained that -- I don't believe that any library fragment that you might be able to incorporate into an application is the same as a Microsoft application. So I really don't think that's a reasonable interpretation of Microsoft application.

Q. Well, is DirectX a Microsoft application under this -- under the plausible reading of 4.A.1. that you and I have been discussing?

A. I'm not actually very familiar with DirectX.

Q. Well, if it's the multimedia subsystem in Windows, is it big enough to be an application?

A. I would imagine that if it's a multimedia subsystem, it probably exposes APIs as a platform for development.

Q. Does that make it middleware?

A. So that would make it middleware. I'm not sure that it would be a natural thing to port, so it may or may not satisfy that definition of middleware, but it might, so DirectX might well be middleware.

Now, I'm not sure that it also satisfies the definition of Microsoft middleware product because, as I said, I'm not very familiar with what DirectX is.

Q. If it provides some of the same functionality as Apple QuickTime for Windows, it would be a Microsoft middleware product, correct, under x.(ii)?

A. Yes, probably, unless it's part of some larger Microsoft middleware product, but it might well be a Microsoft middleware product.

Q. You agree that there are reasons why Microsoft -- legitimate reasons why Microsoft does not want to disclose internal interfaces within blocks of software code that make up the Windows operating system?

A. Yes, that's right. The disclosure of purely internal interfaces might not be a good idea for certain technical reasons.

Q. And one technical reason that would provide a legitimate basis for not wanting to disclose an internal interface is that you might short-circuit certain check routines, privileged checking routines that are important to maintain the stability of the operating system?

A. Yes, in some cases that's right.

Q. And another reason why you might not want to disclose internal interfaces within blocks of software code is that doing so prevents you from rearranging the code inside

those blocks to increase performance or stability or scalability or some other feature over time?
A. That's right.
Q. And you agree that it is possible that there are internal interfaces within blocks of software code defined as Microsoft middleware products that do not have error handling routines that they would need to have if those interfaces were to be called upon by third-party applications or middleware?
A. So you're talking about an internal interface inside a Microsoft middleware product that is not directly called upon from any other Microsoft middleware product or from any other software component outside that Microsoft middleware product?
Q. That is my --
A. That would be an internal interface.
Q. Yes. And you agree with me that such interfaces may not currently have error handling routines that they would need to have if they were to be called upon by third-party applications or middleware?
A. That's right.
Q. Now, when you testified about your second scenario for compliance with Section 1 this morning where the APIs -- where functionality that was part of a Microsoft middleware product gets moved into the core of the operating system

and no longer exposes APIs to applications, does that conflict with Microsoft's obligation to expose all APIs to developers under Section 4.A.?
A. If it's moved into the core of the operating system for use by the core of the operating system so that no -- so that the API that exposes internally to the core of the operating system is not called upon by any Microsoft middleware product or by any Microsoft application, then it is not considered as middleware and Microsoft can make that rearrangement. It would be considered a purely internal API.
Q. Now when you say it cannot be called upon by any Microsoft application, what sort of applications are we now talking about, Word and Excel or the help system of the operating system or both?
A. I think we're talking about Word and Excel.
Q. But it would be all right in your view under Section 1 to move functionality relied on by the Windows help system into the core of the operating system, as long as whatever that functionality is was not exposed through APIs to third-party software developers?
A. That's right.
Q. Now, under 4.C. of the States' proposed remedy, software developers are in certain circumstances entitled to look at the source code of Microsoft operating systems,

correct?
A. That's right.
Q. And what if they, in looking at the source code, discovered, lo and behold, there is all this wonderful functionality in the operating system that isn't exposed to them through published APIs? Can they then start hacking into that code?
A. As a technical matter? As a technical matter --
Q. As a technical matter, first of all.
A. In some cases, it's possible as a technical matter, and in other cases it's not possible as a technical matter, yes.
Q. Would they be entitled to do that under Section 4?
A. Well, under Section 4.C., I believe that Microsoft is permitted to impose terms, some sort of license or nondisclosure agreement that the third party -- that the application developers who visit this secured facility would be required to comply with. And it might be reasonable to impose the terms that they not use this for the purpose of interoperating at any internal Microsoft API, by the definition of internal that we've been using , i.e., not used by some other Microsoft middleware product or application. So that would be perhaps one way to handle this scenario.
Q. You think that would be permissible under Section 4.C.?

A. Yes, I think so.
Q. Now, one of the things that Section 4 requires Microsoft to do, if you look at the embedded definition of technical information, which I call your attention to. It's Paragraph 22.nn. on page 25.
A. All right.
Q. One of the things Microsoft has to do in providing technical information is to provide a reference implementation for its operating systems, correct?
A. Well, for each API and communications interface, Microsoft is required to provide adequate technical information, and in particular under 4.A., Microsoft is required to provide all APIs, technical information, and communications interfaces that Microsoft employs to enable each Microsoft middleware product to interoperate with Microsoft platform software.

So what does that mean that Microsoft employes to enable? Presumably Microsoft documents for the use of its own middleware developers what are the APIs to other parts of the platform software. And in connection with such documentation, it not only lists what the names of the APIs are, it explains how to use them. And there are many ways of explaining how to use an API, and the definition of technical information lists some of the different ways that could be used for a particular API. I imagine that there

is no API for which every one of these kinds of technical information would be useful.

And the limiting thing in Section 4.A. is the technical information that Microsoft employs to enable each Microsoft middleware product to interoperate. So if Microsoft provides a particular kind of technical information to its own developers who have to interoperate across that boundary, then it should provide it to non-Microsoft developers who have to interoperate across that boundary.

Q. Well, if you look at the definition of "Technical Information" in nn, it says in the second sentence, "Technical information includes but is not limited to reference implementations," and then a long series of other things.

Is it your interpretation that despite the presence of the words "includes but is not limited to," that in some instances the technical information required to be disclosed does not include a reference implementation?

A. That's right. If Microsoft or a particular API does not employ a referencing implementation to enable Microsoft middleware products to interoperate with Microsoft platform software or the other kinds of interoperation listed in Sections 1 and 3 of 4.A., then Microsoft is not required to provide a reference implementation in that case.

The purpose of each kind of technical information is to adequately document the means of using the API. And sometimes a reference implementation is very useful for that purpose. A reference implementation is an example of what the implementation might be doing, and it's a detailed technical example of an exemplary way of achieving a certain kind of functionality; it's not the particular software source code that is used to achieve that way.

Q. And in many instances outside of academia where people are really building products to sell, there is no reference implementation other than the shipping product, right?

A. That's right. There are many APIs where it's not necessary to use a reference implementation to adequately document the purpose of the API, so in those cases, there won't be one.

Q. And it's your understanding under Section 4.A. that Microsoft is not obligated to create reference implementations that do not exist?

A. I think that's right. I think -- I think that if they -- the more they adequately document their APIs, the less there will be people visiting the secured facility to try to understand how to interoperate, and Microsoft might choose to provide better technical descriptions of its APIs to lessen the burden of having visitors to its secured facility, and all the kinds of technical information in

that definition are examples of how Microsoft can do that.
Q. Now, going back to Section 4.A. and putting aside for the moment national security concerns and concerns about export control, you believe that if the People's Republic of China has a minister whose job it is to clone Windows, he is a person entitled to disclosures under Section 4.A.?
A. Can you explain what --
MR. HODGES: Objection to the form of the question. It's overly limited. You can't set aside U.S. law and then ask if a representative of another company can come in and pirate the information.
THE COURT: If that's correct, then why don't you reformulate it.
BY MR. HOLLEY:
Q. Could a minister of the People's Republic of China charged in the national interest of his country with cloning Windows view all of the technical information that Microsoft would be required to disclose under Section 4.A.?
A. Well, I'm not sure what you mean by cloning Windows. Do you mean to create a functional substitute for?
Q. That's a very good definition.
A. Okay. Yes, I believe that Microsoft must broadly disclose the interoperability information in Section 4.A., and that people in China will generally be able to read it probably on the Microsoft Web site or in whatever means

Microsoft chooses to disclose it, as they already read similar kinds of information that Microsoft has already disclosed in order to enable the application developers to interoperate with the Microsoft platform software.
Q. If Microsoft invited ten leading software developers to review the early specification of a brand-new operating system that was still on the drawing board in Redmond to find out whether those software developers thought that Microsoft was building a product that they wanted and needed, you don't know whether that disclosure would trigger an obligation under Section 4.A. to provide that same information to the entire world?
A. Are you saying that these ten people are representatives of ISVs?
Q. Yes, sir.
A. So, I think you're asking me about the definition of "timely manner" referred to in definition -- in Section 4.A..
Q. That could bear on your answer, sir, yes. And if you want to look at it, the definition is pp on page 25 of the non-Settling States' proposed remedy.
A. And are you saying that this discussion with the ten representatives of ISVs takes place earlier than the time that this information is disclosed to Microsoft's application developers?

Q. Yes. Hasn't been disclosed to anyone but the people within the Microsoft operating system development group who are drafting a specification for a brand-new operating system.
A. And is it the case that these ten people would be under some sort of nondisclosure obligation to require them not to use this information immediately in building applications? Are they reviewing it for the purpose of commenting on it or for the purpose of getting a head start developing products?
Q. For the purpose of commenting on it.
A. And they are under a nondisclosure obligation not to disclose it further?
Q. Well, they will be but for this decree. Let's assume that --
A. Okay. Then, I think that the only term in the definition of "timely manner" that might be implicated here, of course, is Roman numeral III, "disclosed to any third party," and I think that in that case, one reasonably might not count this as a form of disclosure. If these are people employed as consultants to the point that they are under a very strict nondisclosure, then I think that they almost don't count as a third party, but at this point, you know, it may be beyond my technical expertise as a computer scientist to talk about this kind of business relationship.

Q. The view that you just expressed, I take it, is newly formed over the last six weeks?
A. Well, you did ask me about this question at my deposition, and I guess I've had a chance to reflect on it since then. I don't think I devoted a great deal of thought to it in the meantime.
Q. Okay. And if these people are not consultants to the Microsoft Corporation but rather employees of Lotus and Novell and Borland and Corel, does that alter your analysis?
A. I think if they're under such strict nondisclosure that they can't even disclose it to other employees of Novell and Corel and so on, then they are, in fact, acting as consultants to Microsoft, but again, this may be beyond my expertise as a computer scientist to judge this kind of business relationship.
Q. Under Section 4.C., which has to do with access to Microsoft source code, if I am a 16-year-old living in Tuscaloosa, Alabama, developing software in my garage, I have a right to come to Microsoft's headquarters in Redmond, Washington, under Section 4.C., and look at the source code for Microsoft operating systems?
A. Not necessarily. I think that -- let me turn back to 4.C.. 4.C. has the term "reasonable access"; licensees, third-party licensees and so on shall be permitted

reasonable access to study, and in the explanatory rationale that the States have provided, it's not in this copy of the proposed judgment, they give an example of reasonable access. The example they give is if a certain person has a history of software piracy, Microsoft might deny access.

But presumably that's not the only example. Examples are meant to show that Microsoft has some discretion in good faith, to in good faith deny access, and that discretion might even, probably does even extend to asking the applicant for access which part of the API is unclear that they wish to interoperate with, and so on.

So, I think that in general, one should broadly disclose APIs and so on to a wide range of software developers. There are many software developers in this country who, you know, don't have the benefit of academic credentials or a bachelor's degree or whatever, who are nonetheless innovative and important software developers, and they all had to get their start somewhere. And so they need to be able to interoperate with other software in the world just as much as anyone else.

But I still think that Microsoft has some discretion under the words -- under the terms "reasonable access," to make this judgment if they do so in good faith.

Q. There is nothing in Section 4.C. that says that a third

party viewing the source code can't take notes while they're doing so, is there?
A. I don't see anything about not taking notes.
Q. And is there anything in Section 4.C. that says that a third party with a good memory can't look at clever algorithms that Microsoft has created to perform particular functions and then use those algorithms in the other party's code?
A. It doesn't say that here. To some extent, algorithms can, of course, be patented, and in that case, the developer couldn't use the algorithm.
Q. Well, presumably that wouldn't matter, would it, under Section 15 of the States' remedy because Microsoft would have to give that person a royalty-free license to all of its patents.
A. Only if the patents are implicated in the communication across the API; if the patent is on some technique that's internal to how a function is performed, not which function is performed or how to talk about that function, then it's internal, and that kind of patent need not be licensed under the terms of the States' remedy.
Q. And if the algorithm wasn't patented, a skilled software developer wouldn't have to memorize all 38 million lines of code in the source code for Windows XP in order to glean useful information from reviewing that source code?

A. Well, I guess I have in the past looked at source code under nondisclosure agreements which prohibit me from using trade secrets that I may have learned from looking at the source code. So, to the extent that those trade secrets are about the interface in the same way as we just described -- that I just described with patents, Microsoft may have to disclose or license those trade secrets. But I think that Microsoft could impose terms in its reasonable access agreement about the appropriation of trade secrets.

Q. How many thousands of people per month would be entitled to come to Redmond to look at the source code for all of Microsoft's operating systems under Section 4.C.?

A. I don't know. I think it would depend on whether the disclosures made in 4.A. are better or worse. If the disclosures made under 4.A. are technically adequate to interoperate, then software developers can rely upon those disclosures.

And to learn how to interoperate with a piece of software by reading the source code for that software is very time consuming and therefore expensive. So software developers naturally prefer to have digested descriptions of how to interoperate as called for in 4.A., and to the extent that Microsoft can do that well, then there will be much less need for people to visit the secured facility under 4.C..

Q. Now, one of the things that 4.C. says is that somebody can come study the source code at Microsoft in order to interoperate, and that's a capitalized defined term, correct?
A. Yes.
Q. And the definition of "interoperate" -- maybe we should look at that. That's paragraph 22.Q. on page 22, so Section 22.Q., page 22.
It says that: "Two products must be able to support the full features and functionality of one another." And that's just one way of saying that the two products must be functional substitutes for one another, correct?
A. No, not at all. It means that if one product can do six different things, let's say there's a Microsoft product software, maybe it's on a different -- maybe it's through a communications interface or API, it can do six different things, and another product whose purpose is different, for example, which is connected to it, can ask it to do those six different things.
Now, if the other product that it's connected to is only told how to ask it to do four different things, then it's not accessing the full functionality of the other and it's not able to utilize the full features and functionality of the other.

And if the other product, you know, maybe the one that can do 12 things, isn't able to communicate -- I guess I'm getting too confusing here.

So that's the point of access, utilize, and support: It's being able to use the interface, the API or the communications interface to talk about and request the different functionalities that may be available.

Q. Well, this is a pretty short definition. It says: "Interoperate means the ability of two products to effectively access, utilize, and/or support the full features and functionality of one another."

Where are you deriving this number of or this definition? I mean, do you see that in the words here, sir?

A. Yeah, I'm giving an example. I'm talking about features, for example, the ability of a piece of software to do six different things. That's maybe six different features, all right? And if you only disclose to me the words to ask for four of those features and you don't tell me what words to use to ask for the other two features, then you're not permitting me to interoperate according to the States' definition. There may be some partial interoperation. I can access or utilization some of your features, but not the full features, not all of the features.

Q. Let's pretend that you're computer A and I'm computer B --
A. All right.
Q. -- and we're going to interoperate, and you speak Sanskrit, English, and German, and I speak English and French.
A. All right.
Q. Now, the fact that I don't speak one of the languages that you speak denies us the ability to fully interoperate under this definition, does it not?
A. I guess if you speak English and French, you should disclose how to speak English and French, right, if I wished to interoperate with you, and the idea is that -- let's turn to the use of the word "interoperate" back in -- Now I lost track of where we are. Which provision of the remedy are we at?
Q. We're back in 4.C..
A. All right. Right. If you speak English and French, and you know how to do six different things, presumably six different things unrelated to speaking languages, and I were to teach you German, you would still only know how to do those six different things, right?
The full features referenced here isn't about how to ask for things, it's what things you know how to do, and so a non-Microsoft developer who wants to make a software

product that interoperates with the Microsoft software product needs to know how to ask for all the things that the Microsoft software product knows how to do and is willing to do when another Microsoft product speaks to it.

That's what's meant by interoperate here, and I think that this definition captures that in a reasonable and concise way.

Q. In forming your view, have you reviewed any of the submissions made by Novell, Sun Microsystems, or the IBM Corporation in a proceeding in Brussels in front of the European Commission which centers on the word "interoperate?"

A. No, I have not.

Q. You do not know, do you, whether the .NET framework makes any calls to interfaces of Windows that are not documented in MSDN?

A. That's right.

Q. And you do not know whether there are APIs exposed by the .NET framework that have not been documented for use by software developers?

A. That's right. The .NET framework is a relatively new thing, and I actually have studied parts of it in fair depth, but I don't know about the internals of the Microsoft implementation of it, so I don't know whether it -- whether the Microsoft implementation -- how it calls

upon the underlying Microsoft platform software. And again, since I haven't studied the Microsoft implementation of that framework, I don't know what it might expose to software developers that are different from what's documented. I've read the documentation.

Q. You are familiar with the common language infrastructure because there are other implementations than Microsoft's in existence, correct?

A. There are other implementations than Microsoft being worked on. I don't think that they're at all complete, so in partial existence, yes.

Q. And don't tell me which ones you're aware of, but as far you know, the people that you are aware of that are working on those implementations are continuing to do so?

A. That's right.

THE COURT: If this is a good place to stop, we can -- we can take our afternoon break.

MR. HOLLEY: Yes, Your Honor. Thank you.

THE COURT: All right, we'll take a 15-minute break. So we should be back at quarter of, and we'll resume at that time.

(Thereupon, a break was had from 3:32 to 3:58 p.m.)

THE COURT: All right, good afternoon again.

MR. HOLLEY: Good afternoon, Your Honor.

THE COURT: Let's proceed.

BY MR. HOLLEY:
Q. Professor Appel, the third provision of the non-Settling States' remedy that you were opining about is number 16; is that correct?
A. That's right.
Q. And in paragraph 143 of your written direct testimony which appears on pages 54 and 55, tell me when you're there, sir.
A. Yes.
Q. One of the things you say on the carry-over part of the paragraph on page 55 in the first complete sentence is that "Microsoft can and has subverted reliance on industry standards by not abiding by those standards." Is that your testimony, sir?
A. Yes.
Q. Now, when I asked you at your deposition about manipulation and pollution of industry standards by Microsoft, you told me that what you were relying on was Microsoft's Visual J++ development tools that in your understanding misled developers into writing Windows specific Java applications, correct?
A. I believe I may have said that, yes.
Q. And you agree that the Court of Appeals in this case said that it was perfectly all right for Microsoft to develop a Java run-time environment that did not conform to

Sun's specifications?
A. That's right.
Q. And you also agree, do you not, sir, that if software developers using Visual J++ did not use Microsoft's key words and compiler directives, they could use Visual J++ to write Java applications that could be run on other Java run-time environments?
A. I think key words and compiler directives were one part of the problem, and the other part may have been non-standard class libraries.
Q. But you do agree, sir, that if developers used Microsoft's Visual J++ tools and did not use the key words and compiler directives that call directly to Windows, they could write portable code in Java?
A. Yes, it is possible; it was possible to use the Visual J++ in a mode where one could with care develop portable applications.
Q. And software developers did not have to use Visual J++ at all because there were products from Symantec, Borland, and other suppliers that they could use to write Java applications that could run on Microsoft's Java virtual machine, correct?
A. That's right. I believe the issue was more that Microsoft advertised Visual J++ as a Java compliant or Java standard and so therefore, as a way to develop portable

applications.
Q. Now, with regard to the authorization data field in the Kerberos specification, you agree that Microsoft's use of that, what's sometimes referred to as off data field in its Kerberos tickets, did not prevent the interoperation of Microsoft's implementation of Kerberos with other implementations of Kerberos with regard to authentication as opposed to authorization?
A. That's right. The features of -- the standard features of Kerberos, the Microsoft and non-Microsoft servers and clients, all of the standard features of Kerberos, the Microsoft version of that standard supported for interoperation. It was when a non-Microsoft operating system wished to access some of the additional features that Microsoft's own operating systems supported that Microsoft did not disclose the communications protocol information necessary for full interoperation.
Q. And you agree with me, do you not, Professor Appel that authentication is the principal subject of the Kerberos protocol?
A. I believe that may be the case, yes.
Q. You are not aware, are you, sir, of any software developer in the world who was misled into using the authorization data field in Microsoft Kerberos tickets when they did not want to do so?

A. That is right.
Q. Now, as to HTML extensions, you are aware, are you not, sir, that both Microsoft and Netscape extended HTML in ways that permitted the creation of Web pages that could not be properly displayed in the other Web browsing software?
A. That's right.
Q. And you do not know to what extent Microsoft submitted its extensions of HTML to industry standards bodies like the Internet Engineering Task Force or the Worldwide Web Consortium?
A. That's right.
Q. With regard to Java as an industry standard, you agree with me, do you not, sir, that many aspects of Java are defined by Sun Microsystems?
A. Yes.
Q. And you also agree with me, sir, that Sun Microsystems controls the test suites used to determine whether a particular implementation of a Java run-time environment is compliant with a Java specification?
A. I'm not sure that that's the case, but I would have no information that would contradict that.
Q. Well, look at your deposition, sir, volume 1, page 188, line 10. Do you remember, Professor Appel, being asked the question: "Sun controls the tests that determine whether a particular implementation is compliant with Java, correct?

"Answer: I believe that's right."

Do you remember being asked that question and giving that answer?

A. Yes, clearly I wasn't quite sure then either.

Q. CIFS stands for the Common Internet File System, correct?

A. I'm not actually sure what CIFS stands for.

Q. SMB stands for server message block?

A. SMB I have, you know, used a lot, but more as an acronym than remembering what it stands for. I understand it's SAMBA.

Q. And SAMBA, S-A-M-B-A, is an open source product that implements the SMB protocol on non-Microsoft server operating systems, correct?

A. That's right.

Q. And you, yourself, sir, have used SAMBA for many years, have you not?

A. Yes, I have.

Q. The computer science department at Princeton University uses SAMBA to enable Windows client computers to access files that are stored on non-Microsoft server operating systems, correct?

A. That's right.

Q. And you yourself, sir, use SAMBA currently to access files from your Windows 2000 Professional client on

non-Microsoft servers?
A. That's right; basic file access works fine.
Q. Now one of the provisions of the SRPFJ that you opine on in your written testimony is Section Roman III.J.1., correct?
A. III.J.1.
Q. That's what's commonly referred to as the security carve-out?
A. Yes.
Q. Would it be helpful to you, Professor Appel, to have a copy of the SRPFJ up there with you? I think you may have one, sir, but I'm happy to give you another one.
A. I'm not sure if it's here, and it would be helpful.
MR. HOLLEY: May I approach the witness, Your Honor?
THE COURT: Yes.
THE WITNESS: Thank you.
BY MR. HOLLEY:
Q. And my question just is, is one of the provisions of the SRPFJ that you address in your written direct testimony Section III.J.1?
A. Yes, it is.
Q. Now, you are aware, are you not, sir, of computer scientists in this country who believe that the less information potential hackers have about the manner in

which security is provided by an operating system, the less likely those hackers will be able to break those security mechanisms?
A. I certainly read a statement like that in the expert report of Dr. Bennett, and he is a computer scientist.
Q. You are unaware, sir, of any instance in which Microsoft has failed to disclose the information that other software products need to process security keys generated by Windows operating systems?
A. I believe there has been testimony about Microsoft's nondisclosure to ReaLNetworks of information needed with respect to the secure audio path. I'm not sure specifically with whether that's with respect to keys, but it was about interoperation and an API where there were -- where Microsoft claimed there were security-related issues. Indeed the secure audio path does have some security-related issues.
Q. Okay. But my question, sir, was related specifically to keys. You -- as you sit here today, you're unaware of any instance in which Microsoft has failed to disclose the information that other software products need to process security keys generated by Windows operating systems?
A. Um, I'm not sure. In some sense, the off data field is related to information need to do process security keys, but I'm not aware of large numbers of such instances in any

case.
Q. Your testimony is that the privilege access certificates in Microsoft's Kerberos tickets are security keys?
A. Yes, in the sense that keys and tokens and tickets have similar kinds of functionalities with respect to APIs.
Q. Professor Appel, there is nothing that prevents any other company, in addition to Microsoft, from creating its own digital rights management software, correct?
A. That's right, there are different companies that are creating digital rights management software.
Q. So, there are already multiple kinds of digital rights management software available in the world?
A. Yes.
Q. Now, you are aware that a hacker, an anonymous hacker, has reverse-engineered the digital rights management software in Windows XP, are you not?
A. Yes.
Q. And we discussed at your deposition that document, which is an article which describes in considerable technical detail how the digital rights management software in Windows XP works, correct?
A. That's right.
Q. And by using the information contained in that article, as well as the source code that is referenced in a

hyperlink in that article, someone could steal copyrighted content belonging to media companies like Sony and Bertlesmann and Vivendi by defeating the digital rights management software in Windows XP, correct?

A. That's right. My point in using that example is not that defeating security is a good thing; it's that security is not preserved by hiding APIs. In this case, Microsoft did not disclose those APIs, and yet the hacker was able to find out that kind of information without the Microsoft disclosure.

Q. The publication of the source code referenced in that document violates a fall law called the Digital Millennium Copyright Act, correct?

A. The source code is not contained in the document. It's linked by the document.

Q. But --

A. And there are certainly interpretations of that act upheld in court that, under which the publication of that source code violates the Digital Millennium Copyright Act.

Q. Including litigation in which you, yourself, have participated in; is that correct?

A. I served as a witness in that litigation.

Q. And you submitted a declaration in one of those cases where you argued that the Digital Millennium Copyright Act was an unconstitutional violation of the First Amendment to

the United States Constitution, correct?
A. That's right. I think that restriction on the publication of explanations like that is a violation of the First Amendment. That's my personal belief.
MR. HOLLEY: I have no further questions, Your Honor.
THE COURT: All right. Redirect. I'll give you a few moments to set up
REDIRECT EXAMINATION OF ANDREW APPEL
BY MR. HODGES:
Q. Professor Appel, you were asked yesterday, if you recall, whether you had given any thought to how Microsoft could comply with Section 1 of the States' proposed remedy which requires Microsoft to create unbound versions --
THE COURT: You need to have your voice higher.
BY MR. HODGES:
Q. -- which requires Microsoft to create unbound versions of its operating system products. Do you recall that testimony?
A. Yes, I do. I think there are several ways that -- several technical options that Microsoft has available to it at its discretion in complying with Remedy 1. Now, of course, in the case where the different Microsoft middleware products don't really depend on each other for functionality, then it's very easy to make them removable.

In the case where there is some dependence, then one of the options Microsoft has, and I've explained this so I won't go into great detail, is just let the Microsoft middleware product be removable and OEMs might substitute a non-Microsoft middleware to support that purpose, and in any case, even if they don't, Microsoft is not responsible for the removed functionality.

THE COURT: There wasn't a question, you simply directed him to the area. So, perhaps you need to -- on redirect, he's going to direct you to an area and then he's going to ask you a question which is what you should wait for.

So do you want to pick up on your question?

BY MR. HODGES:

Q. When you were asked the question yesterday about whether you had given any thoughts to how Microsoft could comply with Section 1 of the States' proposed remedy, did you ever finish your answer to that question?

A. No, I didn't. I explained one or two of those ways, but I don't think I explained all of them.

Q. Could you explain what ways in your opinion Microsoft could comply with Section 1 of the States' proposed remedy?

A. One way is to simply let the Microsoft middleware product be removable. Another way is to let subcomponents of the Microsoft middleware products be removable. The

States' remedy doesn't require that, but it permits that.

And then in the case of, for example, MS HTML, the rendering engine the subcomponent of the browser, an OEM might choose to leave that component in even if they want to substitute a different browser, and then there's no chance of degradation of the functionality of other components that depend on that HTML rendering.

Another option, as I have explained, is to take necessary fragments of functionality and embed them in other products, other than Microsoft middleware products, so they don't expose APIs.

Another kind of way to comply is just to reduce the inherent commingling, or I should say interdependence between the Microsoft middleware products. This would be not really a mechanical engineering task; one requiring some design to make the Microsoft middleware products a little less dependent on each other, and Microsoft might choose to do this, for example, if it doesn't like the other options because it doesn't want to be dependent for functionality on a non-Microsoft substitute.

And finally, because I think that this provision doesn't overly specify how Microsoft is to perform this technical job, there might be other technical avenues that Microsoft can use that I haven't even thought of.

Q. So, did I count four different options?

A. I think I listed four.
Q. Would Microsoft have to employ any particular one of those options that you just annunciated?
A. It could use any one of those options or it could use the different options in different combinations; it could employ different options to the different middleware products; it could think of its own technical means of complying that are not among my list.
Q. How many unbound versions of the Windows operating system product would Microsoft have to create under Section 1 of the States' proposed remedy?
A. For each operating -- for each bound operating system that it distributes, such as Windows XP, or maybe even such as Windows XP Home and Windows XP Professional, it would have to also distribute an unbound version, except of course for Windows 98 and 98SE where it has no obligation, and except for Windows 95, which is an unbound operating system.

So for each bound operating system, one unbound operating system.
Q. Does Microsoft currently distribute Windows 95?
A. I don't know that it does.
Q. How long could Microsoft continue to distribute each of the unbound versions that it creates?
A. If Microsoft makes an unbound version of Windows XP,

then it has satisfied its obligation with respect to Provision 1, and it could continue to distribute that unbound version and the bound version of Windows XP for as long as its likes.

Q. Is there anything in Section 1 of the States' remedy in your opinion that would require Microsoft to stop distributing either the bound or the unbound version of, say, Windows XP on some particular date?

A. I think the second paragraph of Provision 1 says, "With respect to the unbound Windows operating system product." And what I understand that to mean is that for a particular bound Windows operating system product, there is one unbound Windows operating system product, that's why we can say, "With respect to the unbound Windows operating system product." So I don't think it's the case that for Windows XP, for example, there would be one unbound version of Windows XP now and one later.

Q. If within six months Microsoft could create an unbound version of Windows XP that complied with Section 1, would it have to either take it off the market or create some other version of Windows XP in six months under the second paragraph of Section 1?

A. If within six months Windows -- Microsoft produces an unbound version of Windows XP that complies with the second paragraph, that is, that it permits the removal only of the

middleware products identified in definition x.(i), then I believe it has discharged its obligation with respect to the Windows XP operating system.

Q. You provided some testified on a product called Windows XP Embedded. Do you recall that testimony?

A. Yes, I do.

Q. Have you had a chance to examine Windows XP Embedded?

A. Yes, I have. I used this -- the -- well, Windows XP Embedded comes in two parts, really. One is the source code which is the same as the source code for Microsoft Windows XP, because it's really the same operating system, and a target designer tool, and so I have used the target designer tool to experiment with different configurations of the Windows XP operating system.

Q. Perhaps we could back up to a higher level, and if you could just explain what Windows XP Embedded is?

A. Well, Windows XP Embedded is a Windows XP operating system that Microsoft markets for OEMs to use for embedded applications. An embedded application is one that, unlike desktop, sits in the device that the end-user may not even realize has a computer in it, for example, a set-top box on top of a television to do cable TV or a video game console, or, I guess, a toaster.

And for that type of application, Microsoft wants to provide its Windows XP operating system but realizes

that the OEM may not need all the functionality of the Windows XP operating system to run a set-top box or a video game console. So they provide the same operating system files as in Windows XP, but they also provide a tool that the OEM can use to select which of those files they want to use if they are building a set-top box instead of a desktop operating system.

Q. What do you mean it has the same files as Windows XP?

A. Well, the Microsoft documentation accompanying Windows XP says it has the same binary files -- I think it even says 100 percent the same binary files as Windows XP.

Now, a binary file is the software code of the operating system, and so, if it has 100 percent the same binary files, the same software code, then I take it to be the same operating system.

Q. When you say that the licensee of Windows XP Embedded could remove components, does that include removing a Microsoft middleware product?

A. Oh, yes. For a set-top box, you won't need many of the Microsoft middleware products; you won't even need the Windows desktop. The kinds of things that can be removed include not only Microsoft middleware products, but lots of individual fragments.

Q. Can Windows XP Embedded run on a personal computer?

A. Technically it can. The license agreement that

Microsoft provides to OEMs with Windows XP Embedded specifies that it is not to be used to configure XP Embedded to make a desktop operating system even though that would be technically possible.

Q. And how would it be technically possible?

A. One would just use the XP Embedded Target Designer tool by moving the mouse and clicking on which components you want to include, include all the components of Windows XP necessary to make a desktop operating system.

Q. Can XP Embedded run the same applications as Windows XP?

A. Oh, yeah, it's the same computer code. It supports the same APIs, so it can run all the same applications if all of those components are included in the configuration.

Q. Could Windows XP Embedded be installed on a PC without, for example, Windows Media Player?

A. Yes, one could make a configuration using this Target Designer that included all the pieces of the desktop operating system, except the media player, and perhaps the way the target designers are currently built, one would also have to leave out components that depended on the media player.

Q. That's my next question: Do you have a question whether this existence of this product, Windows XP Embedded, could be used by Microsoft to create a version of

the Windows operating system product that complies with Section 1 of the States' proposed remedy?
A. Yes, I think with some minor engineering changes to the XP Embedded Target Designer, that would be a good configuration tool that Microsoft would be able to provide to OEMs for the purpose of readily, so to speak, that OEMs could use to readily remove the Microsoft middleware products that they want to remove.
Q. And what minor engineering changes would those be?
A. Well, I can give some examples. Microsoft, the XP Embedded Target Designer is provided with a list of sample configurations. A configuration is just a list of which components to include, and so they provide a sample configuration saying, "Here are the components you might include for a set-top box," and they provide another sample configuration saying, "Here's all the components you might include for a game console."

And the OEM is expected to start with this sample configuration and say, "Well, for my set-top box, I'm going to adjust the sample configuration by removing this component and adding this component," and so on.

But Microsoft has not provided a sample configuration that corresponds to a desktop operating system with all the components necessary to support the applications for the desktop operating system. It's

possible to build such a configuration. The way one would do it is start with some configuration that lacks lots of components and start clicking. And I experimented with doing this: I started clicking on components that I needed to include to make a full-featured Windows XP, one-by-one, and after an hour or so of clicking on components, I got tired of it.

It would have been much easier if one of the sample configurations that Microsoft included corresponded to the set of components necessary for a desktop operating system. So that's one of the ways in which the Target Designer could be adjusted to be useful for the purposes of complying with Provision 1.

Q. Are there other ways?

A. I think that one of the assumptions built into the Target Designer is that if you're going to include one component, then you also need to include any other component on which it depends, which I think is very reasonable for the purposes of constructing embedded set-top boxes and so on. But the States' remedy Provision 1 requires that you be able to construct configurations where you include this Microsoft middleware product and don't include that Microsoft middleware product, even though this one might call upon that one for some of its functionality.

So that assumption built into the Target Designer would have to be changed so that the Target Designer would permit such configurations.

And another example is the Target Designer does have a way to take a group of subcomponents and say, well, this is a major component and maybe I'm going to include this entire major component. It has a way of drawing boundaries around groups of components, and at the moment, there is no way to draw boundaries around the components that correspond to a particular Microsoft middleware product, and so it would be good to have those boundaries specified and have the Target Designer be able to process them.

Now, these are minor engineering changes to this configuration tool, and I don't think they would be technically very difficult for the engineers who built this configuration tool to make these modifications, so that instead of being a way to select which components you want for building set-top boxes and computer game consoles, it would be a way to select which components you want out of an unbound operating system.

Q. And just so we're clear, could you just tell us simply what the Target Designer is?

A. Well, it's a user interface tool, so you run it and it pops up on the screen, and in one subwindow it lists all

the components that are available; in another subwindow, it lists which components you've selected. You can ask it to check dependencies between components and see if you left anything out.

So it's a way for OEMs to evaluate whether the way they've specified which components they want is a reasonable one and is likely to work, and OEMs can, in fact, even include their own components for interoperation with the Microsoft ones.

Q. And the Target Designer is a tool that's currently distributed by Microsoft with XP Embedded; is that correct?

A. That's my understanding, that that's part of the XP -- I guess what Microsoft provides as XP Embedded to OEMs is the binary code for the operating system, which OEMs are expected to redistribute, and the Target Designer, which OEMs aren't really expected to redistribute; they're expected to use it in selecting which components of the operating system to redistribute.

Q. And the last question about XP Embedded is when the XP Embedded runs on something such as a set-top box, is it using the same binary files that Windows XP uses when it's running on a PC?

A. Yes, it's using exactly the same operating system files or that is, whichever ones of those files the OEM has selected, so really Microsoft is suggesting to use the same

operating system on the desktop, on the set-top box, on the computer game console, and they even suggest that one of the permissible uses of XP Embedded is to make a limited functionality word processing tool or office productivity tool that could even be used on a desktop.

Q. Now, apart from XP Embedded, you've also undertaken a review of the Windows XP source code, and that was discussed in your cross-examination. Do you recall that?

A. Yes.

Q. And have you been able to complete the review of the Windows XP source code in the amount of time that it's been available to you?

A. I've considered several different kinds of investigations that I could make on the Windows XP source code and binary code that was provided, and some of those I've had a chance to complete and some of those I included in my direct testimony.

And other investigations I'd like to make require a fair amount of engineering efforts to construct, measurement tools and so on, so I have some ongoing investigations.

Q. And why is it that you haven't been able to complete your investigation of the Windows XP source code in the amount of time that you've been able to work with it?

A. Well, I've been able to work with it only for a few

weeks, and there are many interesting questions relevant to the two proposed judgements that one might ask in connection with this case, and I -- just a few weeks is certainly not been enough time for me to complete all of those investigations.

Q. How many lines of source code is there?

A. When I exclude all the lines of source code that don't actually do anything because they're just comments and blank lines, I get approximately 39 million lines.

Q. That's a lot of lines?

A. That's a lot of stuff to read.

Q. But I want to clarify, the direct testimony that you have submitted in this case, is that in any way dependent on any examination or review of the source code that is ongoing?

A. No, the direct testimony is based only on investigations that I was able to complete.

Q. Professor Appel, do you recall being asked whether Microsoft could comply with Section 1 of the States' proposed remedy by hiding APIs in its Microsoft middleware products?

A. Yes, I think I was asked about that.

Q. And I want to be clear: Is there a difference between hiding APIs and removing end-user access to a particular Microsoft middleware product?

A. Yes, I think there is a difference. There are two ways that one can access a middleware product: One is through APIs where it's serving as a platform for application development, and one is by the end-user who might click on something on the screen. And if you remove only end-user access, then you could still be leaving all the APIs there that serve as a platform for application development, and to the extent that the States' remedy is concerned with the goal having to do with platforms for applications development, there is a significant difference between removing end-user access and removing or hiding APIs.

Q. Is it your testimony in this case that Microsoft could comply with Section 1 of the States' proposed remedy merely by hiding the APIs of its Microsoft middleware products?

A. No, I think that if there are some components of those products that have functionality needed by other products, then the code that implements that functionality might be put in other products, but I don't think that it would be a reasonable way to comply by keeping the entire Microsoft middleware product, regardless of which parts are actually needed, in specific other places.

Q. Have you had the opportunity to review the slides used by Microsoft in the opening statement in this case that contained an excerpt of your deposition testimony?

A. Yes, I think I saw those slides on the Microsoft.

MR. HOLLEY: Your Honor, I think this is well beyond the scope of the cross-examination. I didn't mention anything about the opening or anybody's slides.

MR. HODGES: Your Honor, presumably the information that was conveyed in opening statement was a reflection of what Microsoft anticipated eliciting on cross-examination.

THE COURT: No, if the slides pertained to a topic that they have covered, then I don't have a problem going into it. If it's a totally different topic that has not been talked about, then I don't think we need to go into it.

MR. HODGES: I intend to show Dr. Appel one page of the opening, and I think it pertains directly to a topic that's been raised in cross-examination.

THE COURT: All right. Then I'll wait to see what you say.

MR. HODGES: All right. May I approach the witness, Your Honor?

THE COURT: Yes.

And this is exhibit what?

MR. HODGES: This is marked as Plaintiffs' Exhibit 1834, Your Honor.

THE COURT: Okay. And has this been admitted, not admitted?

MR. HODGES: I'm merely showing to it Professor

Appel; I'm not seeking to introduce it.

THE COURT: Okay.

BY MR. HODGES:

Q. Unfortunately, these aren't paginated, Professor Appel, but I'm going to ask you to flip through at least two-thirds of the way through to --

MR. HODGES: Perhaps I could approach the witness and show him the page?

THE COURT: Go ahead.

BY MR. HODGES:

Q. Professor Appel, I've shown you one page from the slides that Microsoft used in its opening statement, and it's also up here on the screen for your convenience, and it's entitled: "Must Microsoft Let OEMs Remove Microsoft Middleware Product Code?" Subtitled "Compliance with Section 1." Do you see that page?

A. Yes.

Q. Professor Appel, does this slide accurately characterize your testimony as to whether Section 1 of the States' proposed remedy would allow OEMs to remove Microsoft's middleware product code?

MR. HOLLEY: Objection to the form of the question, Your Honor. This is a direct quotation, so is the question is the quotation accurate?

MR. HODGES: The question is whether --

THE COURT: I think -- well, the question, I guess -- well, this is supposedly taken out of a deposition or whatever it was, okay. It would seem to me that you would show him the whole thing, not what's up there, and ask him if that's his opinion or not.

MR. HODGES: Your Honor, what I'm asking him is -- the question posed is must Microsoft let OEMs remove Microsoft middleware product code to comply with Section 1?

THE COURT: Right, and there's a question and he gives an answer, and so what are you asking him, if that's his question and answer?

MR. HODGES: Let me rephrase the question.

THE COURT: Okay.

BY MR. HODGES:

Q. Is the answer to the question posed in the title of this page, must Microsoft let OEMs remove Microsoft middleware product code, is your answer to that question no?

A. No, that's not my answer.

Q. What is your answer to that question?

A. Microsoft must let OEMs remove Microsoft middleware product code. I have testified that in some cases where there are components or fragments of those Microsoft middleware products, then those particular components or fragments could be moved to the other Microsoft middleware

products that need that functionality, so long as they don't expose APIs, but even in that case, the bulk of the Microsoft middleware product code would be removed.

When I referred to in the quoted question, "that component," I'm not referring to a Microsoft middleware product. The context of that question in the deposition was a reference to the MSHTML component of the Internet browser, the Microsoft Internet Explorer, the Microsoft middleware product.

So when I said in the particular case in the context where it was asked that that component, the HTML renderer could be left somewhere else in the operating system, as long as it didn't expose APIs, I said that's right, and then I believe I went on to explain, but I'm not sure.

Q. Do you see where it says: "Tom Greene answered that question yes," according to this Exhibit 1834?

A. Yes, I see that.

Q. Have you reviewed Mr. Greene's deposition in this case?

A. Yes, I have.

Q. Do you disagree with Mr. Greene on this point?

A. No, I think I agree with Mr. Greene on this point. Mr. Greene says you may need to leave sufficient code behind in order to make a particular OS functionality operate. I don't think that means you may need to leave an

entire Microsoft middleware product behind if the OEM specifies that it must be removed.

Q. So to the extent that Plaintiffs' Exhibit 1834 suggests that Tom Greene answered this question yes and you answered this question no, you will disagree with that?

A. I would disagree. I think we both answered it yes.

Q. And if you look at the bottom line of this page from the Microsoft opening, it says: "States' amended proposed remedy," Section 4.A., Roman numeral i, "Can't Hide APIs." Do you believe there is any disagreement between your testimony and what is stated in Section 4.A., Roman numeral i, of the States' proposed remedy?

A. I think that my testimony is entirely consistent with the States' proposed remedy. The States' remedy does let Microsoft hide all internal APIs. It says here in what's been quoted on the screen: "Microsoft shall disclose all APIs to enable each Microsoft middleware product to interoperate, whether or not to internal APIs, with external APIs of the platform software of the Microsoft middleware product."

So if you take a fragment or component or sufficient code and put it in some other place where it doesn't expose APIs, then that API of that fragment or component will not be an API that is used to enable interoperation between components; and therefore, it's

perfectly permissible to hide it even under Section 4 of the States' proposed remedy.

Q. To be clear, what is an internal API?

A. It's an API that is between different subcomponents of a component and not meant for use by things external to that group of subcomponents.

Q. Would the States' proposed remedy require the disclosure of internal APIs?

A. No, only of APIs used to interoperate, and so the vast majority of the APIs in the implementation of the Microsoft operating system product would not have to be disclosed under either remedy.

Q. Is there any way to measure how many internal APIs there are in a Windows operating system product?

A. Yes, one thing I've done is I've measured how much of the lines of source code, of those 39 million lines of source code, appear to be the kind of source code that just describes APIs, and I found that approximately a quarter of the source code of the Microsoft operating system, that is approximately 10 million lines of code, describes APIs, external and internal, and approximately 440,000 lines of APIs describes external APIs that Microsoft has already disclosed.

And so I believe in my direct testimony, I said that the externally disclosed APIs, the ones that Microsoft

discloses, amount to about 1.2 percent of the entire operating system. They amount to, if I do the calculation right, about four and a half or five percent of all of the APIs.

So, the vast majority of APIs would not need to be disclosed by either remedy, and, in fact, the vast majority, I think, and certainly the majority of the APIs that either remedy would require to be disclosed, are already disclosed by Microsoft so that application developers can make their applications interoperate with the Microsoft platform software.

Q. If that's the case, then why is the disclosure remedy under the States' proposed remedy necessary?

A. Because by selectively withholding certain disclosures, Microsoft can make it much more difficult for independent software developers to achieve full interoperation, and just by withholding a few APIs, Microsoft could, I think, cripple the functionality of applications that -- or middleware that depend on that functionality.

Q. Professor Appel, you were asked in your cross-examination whether it would be possible for third parties to create, replicate or create functional substitutes of Microsoft platform software. Do you remember that testimony?

A. Yes.

Q. Do you have an opinion as to whether, based on the disclosures that would be provided under Section 4 of the States' proposed remedy, whether it would be possible for a third party to make a copy of the Windows operating system?
A. I think those disclosures are not technically sufficient to make a copy, unless somebody could, I guess, walk into the facility and remember 39 million lines of code. But the disclosure would be helpful in making things that are not copies but functional substitutes for the Microsoft platform software.
Q. And what would the purpose be of a functional substitute?
A. Well, I guess one would want to sell platform software to a user who might have otherwise bought the Microsoft platform software, so that it could run the set of applications that the end-user wants to run.

There are some users, for example, who want to run a different operating system because of the applications available for that operating system, let's say Linux. There are some applications and server software that runs very well on Linux, but those users also want to be able to run some of the applications for Microsoft platform software. For example, they want to run Microsoft Office or they want to run non-Microsoft application software, one of those 70,000 applications that can run on the Microsoft

platform.

And now they have the choice, they can buy two computers or they can dual boot their computer or they have a variety of technical choices which aren't very attractive, and if one could support some subset of the Windows APIs, or some subset of the APIs exported by a Microsoft middleware, then some of those applications that run on the Microsoft middleware would also run on the other platform, and so a user interested in those applications would be able to purchase them and run them on the functional substitute or the Windows operating system.

Q. Would the disclosures that would be required by Section 4 of the States' proposed remedy, would that enable a third party to develop a full, functional substitute for the entire Windows operating system?

A. In principle, that would be possible, but they would still have to do the engineering work to actually build it. That is, the disclosures of what functionality is requested at each API then leaves one with the task of building and implementing all of that functionality.

Q. What do you mean to build and implement all that --

A. Well, one has to write all of the source code that does all the operations that's being requested at the API boundary, so, as I explained, there were 39 million lines of source code in the Microsoft operating system, and 10

million lines of them specified to API boundaries, so I believe approximately 30 or 29 million lines of code implements the actual operation of all the components of the Windows operating system.

Q. And would the States' remedy require those 29 or 30 million lines of code to be disclosed?

A. The States' remedy certainly doesn't require those 29 million lines to be disclosed, and it also doesn't require almost all of the other -- the 10 million lines of internal API's to be disclosed. So because the source code of the Microsoft operating system and the technical details of how it achieves its functionality are not required to be disclosed under the Microsoft remedy, anybody who would wish to make a functional substitute for it would have to invent and write it on their own, and that means the engineering of a lot of source code.

Q. So the 10 million lines of code you refer to are both the internal and the external APIs; is that right?

A. There are approximately 10 million lines of APIs, and approximately 9 and a half million lines of them are internal APIs that wouldn't have to be disclosed, and of course, there are the approximately 29 million lines of code that are not APIs and would not have to be disclosed.

Q. So does that get us to about 39 million lines of code that would not need to be disclosed? Am I counting

correctly?
A. Yeah, that's about right.
Q. Professor Appel, you were asked earlier whether the RealPlayer distributed by ReaLNetworks would suffer problems if Internet Explorer was not present on the desktop, I think Mr. Holley's words were, "Would it be in a world of hurt."
Do you have an opinion on whether, in fact that would be correct or not?
A. I believe that RealPlayer or the RealNetworks' media player can actually interoperate with other browsers, and I think I've seen testimony to that effect by the witness, Mr. Green from ReaLNetworks.
Q. I believe that's Mr. Richards from ReaLNetworks.
Do you recall what testimony that was that you were looking for -- that you were looking at?
A. I think it was his written direct where he says that when they run the RealMedia player on the Microsoft operating system, they use Internet Explorer to achieve browser functionality, and when they use that same middleware on a non-Microsoft operating system, they use other Internet browsers to achieve that same functionality.
MR. HOLLEY: And can we get to Mr. Richard's direct testimony? Can you take a look? I don't have the copies to hand out, but this has been admitted as an exhibit

previously. Can you tell me what page you're on?

This was Plaintiffs' Exhibit 1598.

BY MR. HODGES:

Q. If you could read that, Professor, and tell me if that's the testimony to which you were referring.

A. Yes, that's the testimony that I'm referring to.

Q. So based on that, you have an understanding that RealPlayer, in fact, will interoperate with browsers other than Internet Explorer?

A. Yes, when RealPlayer gets services from the browser platform software, it's able to get it from a non-Microsoft browser.

Now, it says here they only do this on a non-Microsoft operating system, but it's clear to me that there's no reason they couldn't also do that on a Microsoft operating system. They've just chosen not to because they've done all the work necessary to get the services from the other browser, and it doesn't seem to have been a problem for them.

Q. So you would disagree with Mr. Holley when he stated that the RealPlayer would not work if Internet Explorer is not present; is that correct?

A. That's right, RealPlayer, according to this testimony, already runs in configurations where Internet Explorer isn't present, and it gets the full functionality that it

needs from another browser.
Q. Professor Appel, you were questioned about whether the cross-dependencies of the Microsoft middleware products in Windows XP could be investigated and determined, and do you recall that testimony?
A. Yes.
Q. Is there a way in which these -- this investigation and determination could be accomplished?
A. Yes. I think that this would be a very feasible engineering task. I think that much of the information may already be in the Windows XP Embedded tool kit, which does tell about a lot of the dependencies between all the software components of the Windows operating system including the Microsoft middleware components.

I can't be sure that all of the dependencies information is there and is correct, so I can think of other technical means that I would use to analyze the dependencies between the Microsoft middleware products, and I don't think it's a trivial task, but I don't think it's at all an infeasible task to do in the matter of a few weeks.
Q. Professor Appel, you were questioned on cross-examination about Professor Felten's investigation in the liability phase of this trial. Do you recall that?
A. Yes.

Q. What was the purpose of Professor Felten's investigation that he undertook in the liability phase of the trial?

MR. HOLLEY: I object to the form of the question. I don't think this witness is in any position to talk about what somebody else's purpose was, especially in light of the testimony at his deposition that he's forbidden to talk to Professor Felten about that subject.

THE COURT: I think you're going to have to reframe it.

MR. HODGES: I'll rephrase the question. I'll rephrase the question.

BY MR. HODGES:

Q. Professor Appel, what's the basis for your understanding of Professor Felten's investigation in the first phase of this trial?

A. It's from reading his testimony of what he did and the context of the case in which he did it.

Q. And based on that understanding, do you know what the purpose of his investigation was?

A. Yes, I think I can determine the purpose.

Q. And what do you understand that purpose to be?

A. I think one of the issues at trial was the binding of Internet Explorer to Windows 98, and there were several different mechanisms used in that binding. One was a

license restriction by Microsoft specifying what OEMs could and could not ship. Professor Felten didn't investigate that as part of a technical investigation; that was a non-technical means.

Another means was that Microsoft artificially removed the end-user's ability to delete the browser from the system: That is, for most software that you have on a Windows operating system, after you install the software, if you decide you don't want it there anymore, there is a control panel to remove it, and Microsoft had disabled that for Internet Explorer in Windows 98.

Another way of binding the browser to the operating system involved the different ways that the browser could be invoked: That is, in the Microsoft Windows 98 operating system, there were many different ways that you could start browsing. You could click on an icon on the desktop and it would open up the browser to view something related to that icon. You could type the name of a Web page into some control on the desktop or in the help system or -- there were just approximately 20 different ways that you could get to browsing when you hadn't been browsing already, and these are ways of invoking the browser.

And there was the concept that the user could choose which browser he wanted to use as his default browser, and he could register that choice with the

operating system and say, "When I start -- when I do one of these things to start browsing, I want to use Internet Explorer or I, in general, want to use Netscape Navigator," and so when the user would click on one of the many ways that would invoke browsing, the Microsoft operating system could look up the user's choice of which browser he wanted to use and start that browser. But the binding aspect was that for many of the ways that Microsoft provided to start browsing, it would not respect the user's choice of default browser. Regardless of what the user had specified as the browser he wanted to use, the Internet Explorer browser would be used.

And so what Felton was investigating was different aspects of the technical means that Microsoft had used to bind the browser to the operating system and were those bindings technically necessary? That is, was there any deep technical reason why these bindings were necessary? So what he did was he made a program that would unbind in these different technical ways the browser from the operating system. He made it removable by the end-user control and he made the operating system respect the user's choice of default browser in all the ways of invoking the browser that he could find.

And I think his focus was less on seeing how many modules of software code he could delete from the operating

system. He was concentrating on the technical means of binding.
Q. Was he attempting to substitute an alternate to the Internet Explorer browser?
A. Well, he showed that you could install an alternate browser. I think he used Netscape Navigator. And he showed that not only could you use the end-user removal mechanism to delete the Netscape Navigator icon from the screen, but that the Netscape Navigator browser could be substituted in such a way that for all these means of invoking the browser, the Netscape Navigator browser would be used instead of the Internet Explorer browser.
Q. And was there a question whether even that could be done with Windows 98 and Internet Explorer?
A. Well, Microsoft had represented that the operating system and the browser were so unitary a product that there was no reasonable technical means of separating and unbinding in ways such as I've described, and so he did a technical experiment to see whether that was true.
Q. And was it true?
MR. HOLLEY: Your Honor, I object to the question. Dr. Appel doesn't know what Microsoft asserted, and I don't think he has a basis to opine about whether that was true or not.
THE COURT: Are you objecting to his last answer?

MR. HOLLEY: Yes, I move to strike it.
THE COURT: All right, I'll consider it.
MR. HODGES: I'm sorry, Your Honor.
THE COURT: It seems to me that I'm not sure what -- I don't know whether he knows what Microsoft -- what Microsoft represented, so I would have to have, frankly, some different foundation. So my inclination is not to go by what he has said that they've represented. Presumably the record is there and I can take a look at it as opposed to relying on what he thinks is represented in it.
MR. HODGES: And my intent is simply to ask him his understanding of what the purpose of Dr. Felton's investigation was and what it accomplished.
THE COURT: All right. Well, that we have on the record at this point.
MR. HOLLEY: Well, Your Honor, I would just point out that the this subject was addressed by the Court of Appeals and its conclusions were exactly the opposite of the direction of Mr. Hodge's questions, so I object to this entire line of questioning that its' inconsistent with the Court of Appeals' opinion.
MR. HODGES: Can I move to strike Mr. Holley's testimony, Your Honor?
THE COURT: I don't know that I consider it

testimony. I will take a look at it and go back, and I'm not going to get into an argument with it now, and it is also after five, so where are you in terms of your --

MR. HODGES: May I ask simply two more minutes' worth of questions, Your Honor, and then I'll be finished with Professor Appel.

THE COURT: I will time you for the two minutes. Go ahead. If it's two minutes, I'll let you.

BY MR. HODGES:

Q. Professor Appel, you were questioned about whether the Windows desktop had to be removable under the States' proposed remedy, and I must admit that I didn't quite understand your answer to that question, so can you please tell me, what is your opinion on that issue?

A. The question was whether it might be in some circumstance considered as middleware and therefore whether the Provision 1 might somehow apply to it, but I believe that the Windows desktop -- Microsoft does not need to make the Windows desktop removable under the States' remedy and that's because of Provision 2. In Provision 2, C3 near the bottom of page 4, "Microsoft shall not restrict the OEM from displaying any non-Microsoft desktop, provided that an icon or other means of access that allows the user to access the Windows desktop is also displayed."

And that means that the Windows desktop, the

software that supports the view of icons and Windows and so on and a task bar at the bottom must be present, if Microsoft requires it, on all copies of the Windows operating system shipped by OEMs. So it is clearly not an optionally removable component, and therefore, all of the core Windows APIs upon which the Windows desktop relies are clearly also not optionally removable.

Q. Final question, Professor Appel: You were asked if software that exposes only one API could be middleware. Do you recall that?

A. Yes.

Q. Can you explain how it is that software that exposes only one API can be considered middleware?

A. An API properly speaking is a collection of functions. It may be a small collection of ten functions or one function, or it could be a collection of a thousand functions, and when an application is built to get some of its services from a middleware and other of its services from an underlying operating system, then to the extent that it gets more of its services from a middleware, then it will be that much easier to port to other operating systems. It's not an all-or-nothing thing.

And also, several middleware products can collectively provide enough APIs so that an application by relying on a combination of these middlewares will be

easily portable to another operating system if all of those middlewares are ported. So it shouldn't be the idea that to be middleware it has to provide a complete platform all by itself for the application.

MR. HODGES: Thank you, Professor Appel. I have no more questions. Thank you, Your Honor. That was probably slightly longer than two minutes.

THE COURT: That's fine. I'll excuse you at this time.

If I could just clarify, Mr. Holley, when you were talking about the Court of Appeals, precisely what subject, since he talked about two sets of subjects in his questioning? One was the issue around the end-user and the other was the issue -- is that what you were referring to?

MR. HOLLEY: I was referring in particular to the notion that the Court of Appeals said that it was perfectly all right for Microsoft to override the user's specification of Netscape Navigator as the default Web browsing software in the situation of Windows update, Windows help, and also the in-place navigation in Windows Explorer between Web sites on the one hand and local resources on the other. And do we have a copy of the opinion?

THE COURT: I have it here. That's what I was looking at.

But you're talking about in the context of the either the browser -- the section on browsers or the license restrictions?

MR. HOLLEY: I think in this particular instance, the Court of Appeals is talking about the design of the operating system such that it overrides the user's default choice.

THE COURT: Right. They had it set up and I --

MR. HOLLEY: Your Honor, if you look at 253 F. 3rd on page 67, the Court of Appeals is talking about, "As for the other challenged act that Microsoft took an integrating IE into Windows causing Windows to override the user's choice of a default browser in certain circumstances, Microsoft argues that it has valid technical reasons," and then it goes on to quote Microsoft's brief on this subject.

And then the Court of Appeals concludes that the plaintiffs bear the burden not only of rebutting the proffered justification, but also of demonstrating that the anticompetitive effect of the challenged action outweighs it. In the District Court, plaintiffs appear to have done neither, let alone both. In any event, upon appeal, plaintiffs offered no rebuttal whatsoever. Accordingly, Microsoft may not be held liable for this aspect of its product design."

And the section of Microsoft's appellate brief that

is quoted refers to Windows help, Windows update, and the ability in something called Windows Explorer or My Computer to look both at Web pages and the C drive of your computer in the same window without launching a separate Web browser. So that's what I was referring to, Your Honor.

THE COURT: Anything that you want to add?

MR. HODGES: Yeah, Your Honor, if I may. The purpose of the questioning was not -- certainly not to contradict the Court of Appeals, nor is it to try to allege or prove something other than what was found in the liability phase of the trial.

There was a suggestion in cross-examination that what Professor Felten had undertaken to do was to somehow perform a test that would support or not support Section 1 of the States' remedy about unbinding and commingling. I was simply trying to make the point that the purpose of Professor Felten's examination was for something different, because at that time, Internet Explorer could not be removed from Windows in the usual manner, which is a finding in the case, and he was testing to see whether a substitution could be made.

So he was not trying to remove the code; he was not trying to support Section 1 of the States' proposed remedy which, in fact, didn't even exist at the time. So I -- so I just want to make clear that the purpose of this

testimony -- of Dr. Felton's examination is not the same as what I believe at least was represented in cross-examination.

THE COURT: All right. I'll have to say it got murky in the presentation of it, and you began to lose me near the end. Okay. It's as I recollect. I just wanted to make sure since they discussed two different -- his testimony related to two topics, as to what exactly you were referring to.

MR. HOLLEY: Your Honor, I don't want to belabor this, especially given the hour, but I want to note for the record that I vehemently disagree with what it was that Mr. Hodges says Professor Felton was attempting to do. I don't think now is the occasion to debate it, but I didn't want to let the past that -- to suggest that I agreed with that characterization.

THE COURT: I assumed that you didn't.

MR. HODGES: And I agree it's not the time to argue it.

THE COURT: All right. So tomorrow, then, we start with Mr. Shapiro. All right. Let me, if you will come back for a quick second, I will indicate where you are on your time. So let me excuse you, and I'll see you tomorrow at 9.

(Proceedings adjourned at 5:15 p.m. until Thursday,

April 11, 2002.)

C E R T I F I C A T E

I, Scott L. Wallace, Official Court Reporter for the U.S. District Court for the District of Columbia, do hereby certify that as such reporter I took down in stenotype all of the proceedings had in said U.S. District Court in the above-entitled cause; that I have transcribed my said stenotype notes into typewritten form, as appears in the foregoing Transcript of Proceedings; that said transcript is a complete record of the proceedings had in the trial of said cause and constitutes a true and correct Transcript of Proceedings had therein.

Scott L. Wallace, RDR, CRR
Official Court Reporter

I N D E X

E X H I B I T S

	PAGE