Appeal No. 04-1234

_____

IN THE
UNITED STATES COURT OF APPEALS
FOR THE FEDERAL CIRCUIT

_____

EOLAS TECHNOLOGIES INCORPORATED and
THE REGENTS OF THE UNIVERSITY OF CALIFORNIA,

Plaintiffs-Appellees,

v.

MICROSOFT CORPORATION,

Defendant-Appellant.

_____

Appeal from the United States District Court for the Northern District of Illinois
in case no. 99-CV-626, Judge James B. Zagel

_____

BRIEF OF DEFENDANT-APPELLANT
MICROSOFT CORPORATION

_____

H. Michael Hartmann
Brett A. Hesterberg
Steven P. Petersen
Leydig, Voit & Mayer, Ltd.
Two Prudential Plaza
180 North Stetson Avenue,
  #4900
Chicago, IL 60601-6780
312-616-5600

David T. Pritikin
Constantine L. Trela, Jr.
Richard A. Cederoth
Robert N. Hochman
Sidley Austin Brown
  & Wood LLP
Bank One Plaza
10 South Dearborn Street
Chicago, IL 60603
312-853-7000

Carter G. Phillips
Sidley Austin Brown
  & Wood LLP
1501 K Street, N.W.
Washington, D.C. 20005
202-736-8000

COUNSEL FOR MICROSOFT CORPORATION

CERTIFICATE OF INTEREST

Counsel for Defendant-Appellant Microsoft Corporation certifies the following:

1.   The full name of every party or *amicus curiae* represented by me is:

Microsoft Corporation.

2.   The name of the real party in interest (if the parties named in the caption are not the real parties in interest) represented by me is: Same as stated in paragraph 1.

3.   All parent corporations and any publicly held companies that own 10 percent or more of the stock of the party or *amicus curiae* represented by me are: None.

4.   _X_ There is no such corporation as listed in paragraph 3.

5.   The names of all law firms and the partners or associates that appeared for the party or *amicus curiae* now represented by me in the trial court or agency or are expected to appear in this Court are:

David T. Pritikin
Richard A. Cederoth
William H. Baumgartner, Jr.
Constantine L. Trela, Jr.
Thomas D. Rein
Russell E. Cass
David C. Giardina
Robert N. Hochman
Marc A. Cavan
Sidley Austin Brown & Wood LLP
Bank One Plaza
10 South Dearborn Street
Chicago, Illinois 60603

Carter G. Phillips
Sidley Austin Brown & Wood LLP
1501 K Street, N.W.
Washington, D.C. 20005

Allen E. Hoover
Banner & Witcoff, Ltd.
10 South Wacker Drive, Suite 3000
Chicago, Illinois 60606

H. Michael Hartmann
Brett A. Hesterberg
Steven P. Petersen
Vladan M. Vasiljevic
Leydig, Voit & Mayer, Ltd.
Two Prudential Plaza
180 North Stetson Avenue, #4900
Chicago, Illinois 60601-6780

T. Andrew Culbert
Microsoft Corporation
One Microsoft Way
Building 8, N/203
Redmond, Washington 98502

Dated: June 3, 2004   Respectfully submitted,

                                              _____
                                              Constantine L. Trela, Jr.

TABLE OF CONTENTS

TABLE OF AUTHORITIES

CASES

STATUTES AND RULES

OTHER AUTHORITIES

STATEMENT OF RELATED CASES

No appeal in or from the same civil action was previously before this Court or any other appellate court. There is no case in this or any other court known to directly affect or be directly affected by this Court's decision in this appeal.

A Director-ordered reexamination proceeding is presently pending in the Patent and Trademark Office in which the patentability of the asserted claims of U.S. Patent No. 5,838,906 over the prior art is at issue. 10/30/03 Director Initiated Order for Reexamination, Control No. 90/006,831 (A2525-37). In a first Office Action in that proceeding, the Examiner found all of the claims unpatentable under section 103. 2/25/04 Office Action in *Ex Parte* Reexamination, Control No. 90/006,831. A final determination in that proceeding that the asserted claims are not patentable would moot the infringement claims asserted in this action.

In addition, three other cases pending in this Court, *AT&T Corp. v. Microsoft Corp.*, Appeal No. 04-1285, *Pellegrini v. Analog Devices*, Appeal No. 04-1054, and *NTP, Inc. v. Research In Motion Ltd.*, Appeal No. 03-1615, involve the interpretation and application of 35 U.S.C. § 271(f), an issue presented in this appeal.

JURISDICTIONAL STATEMENT

The district court had jurisdiction under 28 U.S.C. §§ 1331 & 1338.

The court entered final judgment on January 15, 2004. Microsoft filed its notice of

appeal on February 12, 2004. This Court has jurisdiction under 28 U.S.C.

§ 1295(a)(1).

QUESTIONS PRESENTED

1.   With respect to Microsoft's invalidity defenses:

     a.   Whether the inventor of the prior art ViolaWWW
browser ("Viola") "abandoned" his invention as a matter
of law by continuing to refine the code that evidenced
that invention?

     b.   Whether demonstrating to members of the Web
community the Viola browser, which met all limitations
of the '906 claims, could be an invalidating public use?

     c.   Whether Microsoft presented sufficient evidence to
submit its Viola-based anticipation and obviousness
defenses to the jury?

2.   Whether a new trial should be granted on inequitable conduct

based on the inventors' knowledge of and intentional failure to disclose the Viola

prior art to the PTO?

3.   With respect to claim construction:

a.   Whether the district court erred by construing the claim limitation "executable application," contrary to its ordinary meaning, the specification, and the prosecution history, to encompass applications that are not themselves executable?

b.   Whether the district court's instruction erroneously permitted the jury to find infringement even if the operating system identifies and locates the executable application, just as in the prior art distinguished during prosecution and despite the claim language requiring the "browser" to perform those functions?

4.   Whether Microsoft supplies "components" for "combination" abroad under 35 U.S.C. § 271(f) by sending "golden master" disks containing the Windows code to foreign Original Equipment Manufacturers, when the "golden master" never becomes a physical part of any foreign-made product?

STATEMENT OF THE CASE

This suit was filed on February 2, 1999, accusing Microsoft of infringing Claims 1 and 6 of U.S. Patent No. 5,838,906 ("the '906 patent"). The '906 patent claims computer software products and methods of using computers by which users may interact with objects, such as pictures, embedded in Web pages. Plaintiffs alleged that certain features of Microsoft's Internet Explorer incorporate the claimed invention. Microsoft denied infringement and asserted that the claims are invalid and unenforceable.

Plaintiffs' infringement claims were tried to a jury, but the jury was not permitted to consider Microsoft's prior art defenses. On August 11, 2003, the jury found that Microsoft's accused products infringe the asserted claims and awarded royalty damages of $1.47 for each of the 354,124,000 units of Windows with Internet Explorer made and sold worldwide between the patent's issuance on November 17, 1998, and September 30, 2001, an award of $520,562,280. (A47-49.) In a separate bench trial, the district court rejected Microsoft's inequitable conduct defense. (A26-42.)

Microsoft filed motions for JMOL and new trial, which the district court denied on January 14, 2004. (A7-25.) The court awarded $45.3 million in prejudgment interest, increasing the award to more than $565 million, and granted plaintiffs' motion for an injunction, which was stayed pending appeal. (*Id*.) Final judgment was entered on January 15, 2004. (A1.) Microsoft timely appealed.

STATEMENT OF FACTS

A.   The '906 Patent

The '906 patent issued November 17, 1998, from an application filed on October 17, 1994. (A150001.) Only independent Claims 1 and 6 are at issue.

According to plaintiffs, the '906 invention changed the Internet by taking "the Web browser from being able to look at just static Web pages like the

pages in a book to a fully interactive environment on which you can do financial computations, look at news clips, or play games across the Internet." (A100485.) Claim 1 claims a method of using a computer program product — a browser application — in a network environment to interact with "objects" embedded in hypermedia documents, such as Web pages. (A150019-20, col. 16, ln. 62 to col. 17, ln. 28.) According to plaintiffs, the '906 patent's innovation was their browser's ability to use "type information" associated with an object (such as a picture) to identify and locate an "executable application" external to a Web page, which executable application is then automatically launched on a user's computer to display the object and enable the user to interact with it. (A159.) Claim 6 is substantially identical to Claim 1, but claims a "computer program product." (A150020, col. 17, ln. 58 to col. 18, ln. 30.)

B.   Interactivity In The Early 1990s

Despite claiming this technological breakthrough, one of the inventors, David Martin, conceded that by September 1993, when the inventors supposedly conceived their invention, "the idea of having [a] browser use type information to identify and locate [and launch] an executable application" was known in the art, specifically the Mosaic browser prior art. (A100765; A100781; A101199; A100157-58.) Further, there was already another browser — Viola — that taught all the limitations of the '906 claims.

4

Viola: Pei-Yuan Wei belonged to that dedicated group of individuals who helped to develop the World Wide Web. That group collaborated openly in the early 1990s, while the Web was in its nascent stages, to develop technology to make the Web more useful. (A101201.)

Shortly after first being exposed to the Web, Wei developed his Viola Web browser. (A170000-02.) Wei quickly set out to improve his browser so that it could handle interactive content built into Web pages. (A101155; A170003.) Conventional browsers already supported a basic level of interactivity, *e.g.*, Web forms that permitted users to input information. (A101286-87.) Wei sought more, a level of interactivity that would enable a browser automatically to display objects, such as pictures, within a Web page and allow the user to rotate, move, and otherwise interact with them.

By the end of 1992 — well before the '906 inventors supposedly conceived their invention — Wei had improved his Viola browser to permit such interactive display. (A101155-56; A101197; A170005-06.) This browser operated just as the '906 patent describes: by "parsing" (reading textual source code) a "tag" (a coded indicator) in a hypermedia document (the text form of a Web page) which specifies the location of an object external to the hypermedia document,

5

utilizing "type information" to "identify and locate" an "executable application,"[1] then automatically launching the executable application to display the object and allow interaction with it. (A101198.) For example, if the tag called for a picture, an executable application would be launched to draw the picture. (A101198.)

On May 7, 1993, still several months before the alleged conception of the '906 invention, and more than a year before the filing date, Wei demonstrated, without any confidentiality restrictions, a complete working version of his Viola browser to other members of the Web community, specifically engineers from Sun Microsystems ("Sun"), and explained how it worked. (*See* A101162-63; A101176; A170007.) The source code of the version of Viola used in that demonstration (DX34) was preserved and is the oldest surviving version of Viola that includes the capabilities of the '906 patent. (A101156-57; A170022.) Wei continued to refine his browser in an effort to improve it, but at no time did he remove the relevant functionality: the ability to parse a tag in a Web page that specifies the location of an object, and to identify and locate an application that will automatically display the object and allow interaction with it. (A101173-74; A101183-84.)

---

[1] The quoted terms refer to information associated with the "object" that computer software can use to determine what other computer program will display the object.

In response to Sun's request, Wei provided the Viola source code to Sun on May 31, 1993 (again, without any confidentiality restrictions), by posting the code on a publicly-accessible Internet site and notifying a Sun engineer, James Kempf, that the code was available for downloading.[2] (A101198; A101202; *see also* A170024.) Wei had tried to improve this version of the Viola code (DX37) in some respects over the earlier version (DX34), an effort that inadvertently introduced certain bugs. (A101198-99; A101204.) Despite these bugs (which were easily fixed (*see* A101283-84)), Microsoft presented testimony that the May 31 version of Viola, like the May 7 version, taught all limitations of the '906 claims. (A101198-99; A101274-79.)

Wei shared his innovation with others before the '906 inventors claim to have conceived their invention by, for example, communicating with various people in the Web community about his browser. (*See, e.g.,* A170003-06; A170037-43; A170046-77.) And he demonstrated his invention at the World Wide Web Wizards Conference, again without confidentiality restrictions, in July 1993. (A101213-14.)

---

[2] Kempf downloaded the Viola code, and Sun attempted to use it. (*See* A170025-33.)

Wei continued to refine his invention after the '906 inventors' alleged conception date, inviting and responding to suggestions from those with whom he had shared his browser, and responding to changes in the protocols and standards that were evolving for the Web. He released an alpha version of the Viola code to interested members of the public for free and unrestricted use in October 1993.[3] (A101200-01; A170034-36.) He released a beta version in February 1994, again for free and unrestricted use.[4] (A101167-68.) Wei continued to work on Viola until 1996. (A101162.)

Mosaic: The Mosaic Web browser was developed by a team led by Marc Andreessen, later a Netscape founder. (A100721.) With Mosaic, users could go from one Website to another by clicking on links in Web pages. (A100721.) In addition, Mosaic — by launching "helper applications" — was able to display objects, including pictures, and to permit users to interact with them. (A100674; A100721.) Helper applications are standalone applications that allow a user to view and interact with embedded data in a separate window, rather than in the

---

[3] An "alpha release" is a version for specialists to test to provide feedback to the software's author. (A101200.)

[4] A "beta release" occurs after the development stage and allows general users to try new software. (A101167.)

browser window itself as in Viola and the '906 patent. (A100158-59.) As '906

patent co-inventor Martin admitted, Mosaic used type information in this process.

(A100765.) Mosaic's code was freely available for noncommercial use before the

claimed '906 conception date. (A100722.)

C. Development Of The '906 Patent

1. Events Leading To The Patent

The '906 patent names three inventors: Michael Doyle, Cheong Ang,

and David Martin. According to Doyle and Martin, a "definite and permanent

idea" of the '906 invention was formed by September 7, 1993. (A100750;

A100627; *see also* A100717-18.) The inventors used the publicly-available

Mosaic code as the foundation for their browser, adding just 305 lines to Mosaic's

more-than-100,000 lines of code. (A100720; A100722-23.)

The inventors testified that, by late November 1993, they had created

a working prototype (A100752) and had demonstrated it to outsiders (A100636-

38). They testified that further demonstrations followed, in December 1993

(A100638) and January 1994 (A100757). In contrast to Viola, the software code

used in these demonstrations does not exist; the earliest existing '906 software

code was submitted with the patent application in October 1994. (A100725.)

Before filing the application, Doyle publicized the supposed invention

by posting a press release on an Internet mailing list on August 30, 1994, stating

that "[r]esearchers at the U. of California have created software for embedding

interactive program objects within hypermedia documents." (A151483 (emphasis

omitted).) Wei noticed Doyle's posting, and promptly responded to the mailing

list: "I don't think this is the first case of program objects embedded in docs and

transported over the WWW. Viola has had this capabilities [sic] for months and

months now." (A170078.)[5] Wei also provided a link to a Web page where anyone

"interested in learning more about how violaWWW does this embedded objects

thing" could get a paper on Viola. (*Id.*) Doyle promptly obtained the paper and

was sufficiently concerned by what he saw to respond to Wei that same night

(A101671-72; A101737-38), asking Wei "[h]ow many months and months" Viola

had had the capability. (A170082.) Wei responded that "[d]efinitely by May 8,

1993 we had demonstrated that plotting demo (the very one shown in the viola

paper) to visitors from a certain computer manufacturer." (A170082.)

---

[5] This was not the first time Doyle was directed to Viola. Three months earlier,
David Raggett of Hewlett-Packard directed Doyle to a Website with a link to
Viola, and indicated that Viola "provide[d] a level of embedding." (A170044-45;
A101617-18.) Doyle immediately forwarded Raggett's email to co-inventor
Martin. (A170042; A101620-21.)

On October 17, 1994, six weeks after Doyle's exchange with Wei, the

'906 inventors filed their patent application (A150023-64). With it, the inventors

submitted the earliest still-existing version of their code. (A100726-28.)

In August 1995 — while the application was pending — Eolas, which

Doyle, Martin, and Ang had formed, obtained an exclusive license to the patent.

(A100702.) Doyle announced the license on August 21, 1995, by posting a release

on the Internet, stating that Eolas had obtained "exclusive rights to a pending

patent covering the use of embedded program objects, or 'applets,' within World

Wide Web documents." (*See* A1716; *see also* A170086.) Wei, again, quickly

responded by email, stating that "'technology which enabled Web documents to

contain fully-interactive "inline" program objects' was existing in Viola and was

released to the public, and in full source code form, even back in 1993."

(A170086.) Wei also stated that "[a]ctual conceptualization and existence occured

[sic] before '93." (*Id.*) Doyle responded by mischaracterizing Wei's prior

statements about Viola's capabilities in an effort to get Wei to admit that Viola was

not prior art. (A170084-85; A39.) Wei remained firm in his claim of prior

invention. (A170084.)

2.   '906 Prosecution

The '906 inventors never disclosed Viola to the PTO. Still, their

application encountered numerous obstacles. Many of the features that plaintiffs

11

now describe as central to the invention — the use of type information, identification and location of an executable application by the browser, automatic invocation of the application — were not in the original 43 claims. (A150054-61.) These limitations emerged over the course of years of rejections, arguments, and amendments in the PTO.

One key rejection determined that the '906 invention was an obvious combination of Mosaic and U.S. Patent No. 5,206,951 ("Khoyi"). (*See* A150313-16.) Khoyi describes an "Object Linking and Embedding" (OLE)-style system. "OLE" is a Microsoft system that allows a visual object created using one application to be displayed and edited inside the visual context of another application. (A100164-65.) Khoyi teaches an operating system in which functions normally performed by standalone applications can be made into software components[6] that can be reused in many different application programs. (*See* A72.) As Khoyi explains, in the preferred embodiment, the components "are stored in a shared subroutine library [and] are dynamically linked as needed at runtime." (A170476, col. 12, lns. 30-32.) On June 5, 1997, the applicants filed an amendment that included two arguments relevant here.

---

[6] Standalone applications (such as Word and Excel) can run independently of another application, whereas components can run only as part of another already-running application. (A100203-04.)

12

First, the applicants argued that Khoyi used the operating system to link applications with objects, whereas the claimed invention relied on the browser, not the operating system, to identify and locate the applications. The applicants asserted that with OLE "[t]he actual linking operations are coordinated by the operating system." (A150342; *see also* A150340 (asserting that Khoyi disclosed "an operating system based on capabilities similar to OLE, as used for example in Windows 95")).) The applicants stressed that, in Khoyi, the functions are not performed by the browser, but rather by "*a fully-independent and proprietary operating system.*" (A150344.)

Second, the applicants asserted that Khoyi — unlike the '906 invention — used libraries of routines (*i.e.*, components), rather than standalone applications. (A150344.) Specifically, in Khoyi "functions and operations which would normally be performed by the applications programs themselves, are performed by libraries of routines." (*Id.*) The applicants also noted the difficulty of modifying Mosaic to work with Khoyi's components, stating that "Mosaic would have had to be significantly modified in a number of additional complex and nonobvious ways to achieve the combination." (A150345.)

The Examiner withdrew the Khoyi rejection (A150390) but issued a new rejection based on admitted prior art (Mosaic, HTTP, HTML, and the World

13

Wide Web) and U.S. Patent No. 5,581,686 ("Koppolu"). (A150389-95.) Koppolu, which covers aspects of Microsoft's OLE technology (A100700; A100080), allows a user to interact with embedded or linked data in a window created by a first application, called a container application, even though another application, called a server or containee application, displays the data. (*Id.*) The containee application may be a component, such as an "object handler," implemented as a Dynamically Linked Library ("DLL"). (A100164-65.)[7]

The Examiner concluded that "it would have been obvious to enhance Mosaic by providing such a DLL object handler that would be automatically invoked at document rendering time to provide display and interactive processing of the object within a window in the hypermedia document." (A150905.) In response, the applicants again argued that the prior art relied upon the operating system to identify and locate the executable application:

> The actual linking mechanism [in OLE] between the container document and the containee server application *is coordinated by the operating system's registry database....* At the time of initial object selection by the user, and prior to server application launching, *OLE references the operating system's global registry database* in order to identify which server application is related to a particular data object and to determine what

---

[7] A DLL is a component (such as a spellchecker) that can run only within another application. (A100203-05.)

14

interactive operations are provided by the relevant server application.

(A150902 (emphasis added).)

In addition, the applicants, in an interview with the Examiner, initially tried to distinguish Koppolu by arguing that OLE did not provide automatic invocation of any application that allowed *interactivity*. (A150521.) When this effort failed, the applicants distinguished DLLs as a class, asserting that their invention, which used executable applications, was preferable to the prior art use of automatically-invoked DLLs (A150906-07), and that, unlike executable applications, DLLs cannot "ever run stand-alone (like a local server EXE can)" (A150907).

The Examiner allowed the claims. In doing so, the Examiner made clear that the claimed "executable application" was distinct from DLLs, which cannot run standalone. (*See* A151031 ("The examiner agrees that the claimed external executable application is not a code library extension nor object handler (e.g. windows dll and OLE) as pointed out in applicant's argument (paper #19 pages 12-14).").) The '906 patent issued on November 17, 1998. (A150001-20.)

D. Internet Explorer

The accused Microsoft products are versions of Internet Explorer ("IE"), beginning with version 3.0. (A100826.) Microsoft introduced that version

in 1996, more than two years before the '906 patent issued. (A101237.) IE's

support for Microsoft's ActiveX controls, Java's applets, and Netscape's plug-ins,

which all impart interactivity to Web pages, allegedly infringes the '906 patent.

As relevant here, Microsoft argued non-infringement on two grounds.

First, IE does not use an "executable application" as required by the '906 claims

because ActiveX controls, applets, and plug-ins are all components that can run

only within another active application, not standalone. Second, IE relies upon

OLE within the Windows operating system, which was distinguished by the '906

applicants, to determine which application to run and where the relevant code is

located, so the "identify" and "locate" functions are not performed "by said

browser" as the claims require. A review of the technical steps involved in the use

of these components will place the issues in context.

ActiveX Controls: ActiveX controls began as part of OLE and were

originally called OLE controls; they can be used with a variety of applications.

(A101067-69; A101077-80; A101264-69.) When a Web page uses an ActiveX

control, a tag (the "OBJECT" tag) in the page provides information to the

Windows OLE system to identify the ActiveX control, locate that control in its file

system, and launch it. IE encounters the OBJECT tag while parsing the page. In

most cases, the tag contains a hexidecimal number called a CLASSID, but the tag

16

might also simply indicate the type of data object. (A101068; A101081-82; A101268.) IE converts the hexidecimal CLASSID to its binary form and passes that binary CLASSID to the Windows OLE system, which then performs a series of steps to identify, locate, and launch the proper ActiveX control. (A1694.) In essence, just as described in Koppolu, Windows uses the binary CLASSID to find, within the operating system registry, the name and file path of the appropriate ActiveX control.[8] (A1694; A101266.) If the Web page tag does not provide a CLASSID, IE uses the information that is provided to derive a CLASSID, which it passes to Windows, which performs the steps described above. (A101082; A101268.) The operating system then launches the ActiveX control. (A1694-98.)

Applets: Applets are components that run through the use of another component called a Java Virtual Machine ("JVM"). (A101082-83; A101265; A1698.) The JVM is an ActiveX control that is identified and located like any other ActiveX control. (A1699; A101082-83; A101265; A101269; A1733-34.)

Plug-Ins: Plug-ins are software components developed by Netscape (A101083); the tag associated with plug-ins is the EMBED tag. (A100852.)

---

[8] The COM subsystem of Windows performs these identify and locate operations. COM incorporates the relevant portion of the OLE technology. (A100165-66; A170596.)

Although versions of IE prior to IE 6.0 support the use of Netscape plug-ins,

Microsoft has never supplied plug-ins with IE. (A101083.)

E. The Rulings At Issue

1. The *Markman* Rulings

The district court's constructions of two claim limitations are at issue:

(1) "executable application" and (2) that type information be utilized "by said

browser to identify and locate" the executable application.

Executable Application: The claims require that an "executable

application" be automatically invoked to process an embedded data object. (*See*

A63.) As noted above, the applicants secured the '906 patent by making clear to

the Examiner that Khoyi and Koppolu relied on DLLs or other components, while

their invention did not. Before the district court, plaintiffs escaped this limitation,

convincing the court that an "executable application" can include software

components, such as DLLs. (A64.) The court thus construed "executable

application" to include components: "any computer program code, that is not the

operating system or a utility, that is launched to enable an end-user to directly

interact with data." (A79; A89.)

Identify and Locate by the Browser: The claim language is clear that

"the browser" must identify and locate the application. Everyone agreed below

18

that operating systems *always* play *some* role in the operation of any program, including a browser. (A100029; A100312-13.) The parties' dispute concerned the permissible role of the operating system in *the identifying and locating* functions. (A88.) The court recognized at the *Markman* stage that the inventors, to distinguish OLE, had *dis*claimed any role for the operating system in identifying and locating the executable application (A87-88), and that if IE operates like OLE, there is no infringement. (A88 ("the functions of utilizing the type information to identify and locate the executable application must be performed by the browser, not the operating system as in Koppolu's OLE.").)

> The court's jury instruction, however, deviated from this construction:

> Utilized by said browser to identify and locate means that the enumerated functions are performed by the browser. In other words, the browser connects the type information to identify and locate the executable application. Executing the application once it has been identified and located is not part of this linking.

> *The inventors contemplated the browser's use of some outside resources such as the operating system, as operating systems are always involved in the operation of computer programs.* Nevertheless it must be the browser, not the operating system, that must do the heavy lifting of identifying and locating the executable application. In order to determine what constitutes heavy lifting, you must examine the specific facts that have been presented to you during this trial and decide what the browser's utilizing and how it is utilizing it.

(A101463 (emphasis added).) The court rejected Microsoft's request that the jury

be instructed, consistent with the intrinsic evidence, that the claims do not apply

where the "operating system … determines what application will be invoked and

where the executable files for that application are located." (A1800.)

2. Rulings On Anticipation And Obviousness Defenses Based On Viola

Microsoft argued that the '906 patent is invalid because it was

anticipated by Viola under sections 102(a), (b), and (g) or, alternatively, was

obvious under section 103. The district court refused to allow the jury to consider

these defenses. Based on its legal interpretation of section 102, the court excluded

much of Microsoft's evidence. Then, after allowing Microsoft to present only a

portion of its evidence, the court held that evidence insufficient as a matter of law.

Evidentiary Rulings: Microsoft's evidence included a complete and

functional copy of Viola code that Microsoft planned to demonstrate for the jury.

(*See* A170022.) This was the version (DX34) that had been demonstrated to Sun

in early May 1993. Microsoft also offered documents to corroborate Wei's

testimony concerning the demonstration. (*See, e.g.,* A170007; A170008.) And

Microsoft offered the testimony of a Sun engineer, Karl Jacob, who witnessed the

DX34 Viola code in operation, and two of Wei's co-workers as further

corroboration. (A2338-39; A101212-17.) Microsoft demonstrated the May 7 code

outside the jury's presence and showed that it functioned in a manner encompassed by the '906 claims. (A101158-59.)[9]

The district court prevented the jury from learning about the DX34 version of the Viola code. The court ruled as a matter of law that DX34, which Microsoft offered as evidence of Wei's prior invention, was "abandoned, suppressed or concealed" (and thus could not anticipate under section 102(g)). The district court treated each version of the Viola code as a separate prior art reference, rather than as evidence of Wei's prior invention, and concluded that Wei's changes to the Viola code — changes designed to improve it — represented an abandonment of DX34. (A101187-88.) The court ignored evidence that *all* versions of Viola, from May 7, 1993, forward, included the '906 functionality. (A101173-74; A101183-84.) In short, the court implicitly adopted the following remarkable proposition of law: when a software developer reduces an invention to

---

[9] At the time the DX34 code was written, the HTTP protocol for the Web was in transition from HTTP 0.9 to HTTP 1.0, and the DX34 code was written for HTTP 0.9. (A101160; A101177.) Thus, for the DX34 version of Viola to display a hypermedia document retrieved from the Internet, the Internet server from which the document was retrieved had to send the document with an HTTP 0.9 header. (A101177-78.) For its demonstration, Microsoft used a server programmed to send a 0.9 header. (A101160.) For purposes of the '906 patent, it is immaterial what protocol the browser is able to process because the patent merely requires the browser to work in "*a* distributed hypermedia environment." (A150019, col. 16, lns. 66-67; A150020, col. 17, ln. 61 (emphasis added).)

21

practice, he abandons that invention by creating improved versions, even when the improvements do not remove the relevant innovative features.[10]

The district court, with little explanation, applied similar reasoning to reject Microsoft's section 102(b) public use defense. The court held that the May 7 demonstration to Sun was not a public use, in part because Wei later shared different versions of Viola code with broader segments of the public. (A101189.)

The JMOL Ruling: Microsoft thus was left to build its invalidity case with less evidence than it had at its disposal. But what remained was still enough to go to the jury. Microsoft presented the Viola code that had been provided to Sun on May 31, 1993, without any confidentiality restrictions (DX37). (A170023.) Microsoft also presented testimony from both Wei and its expert Dr. Kelly that the DX37 Viola code taught all limitations of the '906 claims. (*See, e.g.,* A101197-99; A101201; A101274-79.)

The district court nonetheless concluded that the DX37 code lacked unspecified "important elements" and, therefore, did not anticipate. (*See*

---

[10] Oddly, the court applied this rule only to Viola, and not to the '906 code. The court deemed November 1993 to be the earliest reduction to practice of the '906 invention, even though that code was changed and never submitted to the PTO. (A35-36; A100725-26.)

22

A101362-63.)[11] The court also rejected Microsoft's obviousness defense as a

matter of law. (A101362.)

Although the court instructed the jury that it was not to consider

anticipation and obviousness (A101431), the jury, almost immediately after

retiring, asked: "Are we supposed to ignore the testimony of Pei Wei and

Dr. Kelly when deciding if Pei Wei was the first inventor and not Doyle?"

(A101468.) The court answered "yes." (*Id.*)

3.   Section 271(f) Ruling

Plaintiffs claimed royalty damages with respect to foreign, as well as

domestic, sales of Windows. The district court allowed that claim based on 35

U.S.C. § 271(f).

Microsoft sends a number of "golden master" disks — disks that

contain the code for the Windows operating system — to foreign Original

Equipment Manufacturers ("OEMs"), who copy that code onto computer hard

drives for sale outside the U.S. (*See* A52.) The allegedly infringing products are

the computers made and sold abroad. The district court concluded that, by

---

[11] The court commented that the case for anticipation "is much stronger" with respect to DX34. (A101362.) Thus, having excluded Microsoft's strongest evidence, the court took the entire defense from the jury based on the view that Microsoft's remaining evidence was insufficient.

shipping the golden masters to foreign OEMs, Microsoft supplies "components of a patented invention" from the U.S. in "such manner as to actively induce *the combination of such components* outside of the United States." 35 U.S.C. § 271(f)(1) (emphasis added); *see also* 35 U.S.C. § 271(f)(2). That is, the court held that a golden master is a "component" of an infringing product (even though the golden master itself never becomes part of the product),[12] and that copying the intangible software code on the golden master onto a computer hard drive is a "combination" of "components."

4. The Verdict, Inequitable Conduct Ruling, And Post-Trial Motions

The jury found that Microsoft's accused products infringe Claims 1 and 6 and awarded damages of $1.47 per unit, or $520,562,280, for the period between November 17, 1998, and September 30, 2001. (A47-49.)

The court thereafter conducted a bench trial on Microsoft's claim that Doyle committed inequitable conduct by withholding information regarding Viola from the PTO. The court found that the information Doyle actually possessed concerning Viola was not material. (A35-38.) Having found the Viola

---

[12] The golden master itself is not an infringing product because the disks are not operable without being installed on a computer. (A51-52.)

information immaterial, the court concluded that Doyle lacked deceptive intent. (A39-41.)

Microsoft filed post-trial motions, which were denied. The district court granted plaintiffs' motion for an injunction, staying its effective date pending appeal, and awarded prejudgment interest. (A18-23.)[13] This appeal followed.

SUMMARY OF ARGUMENT

The district court's multiple errors concerning Viola distorted the proceedings in a profound and fundamental way. First, and most striking, the court excluded from evidence DX34, a fully functional and anticipating version of the Viola code, based on its erroneous view of abandonment under section 102(g) and its application of that erroneous abandonment analysis to Microsoft's public use defense.

The district court ignored the framework this Court has established for determining abandonment under section 102(g). The court erroneously treated each iteration of Viola code as a distinct invention, rather than as *evidence* of the prior invention, and concluded that DX34 was abandoned when Wei improved his browser's code, even though the anticipating features of Viola were never

_____

[13] The court also granted plaintiffs' motion for an equitable accounting to assess additional royalties, but deferred resolution of disputes concerning the permissible scope of the accounting until after appeal. (A24-25.)

removed. This focus on particular code versions has no basis in the law, and would defeat the well-recognized and sound policies advanced by section 102(g).

The court applied its erroneous abandonment analysis to public use under section 102(b) as well. There is no basis for conflating the two inquiries. Driven by its idiosyncratic view of abandonment, the court failed to recognize that the demonstration of DX34 to Sun was an invalidating public use. *See Netscape Communications Corp. v. Konrad*, 295 F.3d 1315 (Fed. Cir. 2002).

Even with the erroneous exclusion of DX34, Microsoft presented sufficient evidence to support its anticipation and obviousness defenses. The court took these defenses from the jury based on its own assessment of DX37, but failed to explain which element of the invention was missing. Microsoft presented substantial evidence that DX37 teaches every element of the claimed invention, and the jury should have been permitted to evaluate Microsoft's evidence. And, even if DX37 does not teach every element of the invention, the jury should have been allowed to consider whether supplying any missing parts would have been obvious.

The court's inequitable conduct analysis was undermined by its erroneous view of Viola. The court again treated particular iterations of the Viola code, rather than Viola itself, as the relevant prior invention and concluded that

Doyle's supposed inability to obtain Viola code that predated his claimed conception date rendered the information he did have about Viola immaterial. The Viola browser was material, Doyle knew it was material, and he intentionally failed to disclose it to the PTO. The evidence compels the conclusion that this failure to do so was the product of an intent to mislead.

The court also erred in construing two claim limitations. First, the court erroneously construed "executable application" to cover software components rather than just standalone applications. This construction ignores the ordinary meaning of the claim terms, disregards the prosecution history (where the applicants distinguished Khoyi and Koppolu based on their use of components — *i.e.*, DLLs), and ignores that *every* embodiment in the specification uses standalone applications. Under a proper claim construction, infringement cannot be shown because there is no evidence that IE uses standalone applications.

Second, the court's construction of the limitation that the executable application must be identified and located "by said browser" allowed the jury to find infringement even if, as in the OLE technology distinguished during prosecution, the *operating system* consults a registry to determine which application to run and where the relevant code is located. This was error because the '906 applicants, to distinguish OLE, asserted that their invention relies on the

browser, not the operating system, to perform these functions. A properly-instructed jury would have found for Microsoft based on plaintiffs' admission that IE uses the operating system to identify and locate the application that displays the object.

Finally, the district court erroneously interpreted 35 U.S.C. § 271(f) to permit a royalty award on foreign sales. Section 271(f) makes it an act of infringement to export U.S.-made components of a patented invention from the U.S. for assembly into the invention outside the U.S. But the "golden master" disks made domestically and sent abroad are not combined with the foreign-made computers, nor are they components of them. Instead, the information on those disks is copied onto foreign-made hard drives. The copying abroad of information compiled in the U.S. is not within the scope of section 271(f).

ARGUMENT

I.   STANDARD OF REVIEW

The following rulings are reviewed *de novo*: the exclusion of evidence based on interpretation of the patent laws, *Sulzer Textil A.G. v. Picanol N.V.*, 358 F.3d 1356, 1363 (Fed. Cir. 2004); the decision to grant plaintiffs JMOL rejecting Microsoft's anticipation and obviousness defenses, *General Elec. Co. v. Nintendo Co., Ltd.,* 179 F.3d 1350, 1353 (Fed. Cir. 1999); claim construction, *Texas Digital*

Sys., Inc. v. Telegenix, Inc., 308 F.3d 1193, 1201 (Fed. Cir. 2002); and the

interpretation of 35 U.S.C. § 271(f), *Doyon, Ltd. v. U.S.*, 214 F.3d 1309, 1314

(Fed. Cir. 2000).

The ruling with respect to inequitable conduct is reviewed for abuse

of discretion; it will be reversed if it rests on clearly erroneous findings of fact or a

misapplication or misinterpretation of applicable law, or if it evidences a clear

error of judgment. *Brasseler, U.S.A. I, L.P. v. Stryker Sales Corp.*, 267 F.3d 1370,

1379 (Fed. Cir. 2001).

II.   THE DISTRICT COURT ERRONEOUSLY PREVENTED MICROSOFT FROM PRESENTING ITS PRIOR ART DEFENSES.

Microsoft's anticipation defense required it to prove, by clear and

convincing evidence, that Viola disclosed "each and every limitation of the

claimed invention." *Schering Corp. v. Geneva Pharms. Corp.*, 339 F.3d 1373,

1377 (Fed. Cir. 2003). In addition, to establish anticipation under section 102(a) or

(b), Microsoft had to prove that Viola was known by others or described in a

printed publication or publicly used before the relevant date. *Mahurkar v. C.R.*

Bard, Inc., 79 F.3d 1572, 1576 (Fed. Cir. 1996); *Tone Bros., Inc. v. Sysco Corp.*,

28 F.3d 1192, 1197 n.4 (Fed. Cir. 1994). As for anticipation under section 102(g),

once Microsoft demonstrated that Viola included every limitation of the claims,

plaintiffs could present evidence that Viola was abandoned, suppressed, or

concealed. If plaintiffs established a genuine issue of fact, Microsoft then must

present clear and convincing evidence that Viola was not abandoned, suppressed,

or concealed. *Dow Chemical Co. v. Astro-Valcour, Inc.*, 267 F.3d 1334, 1339

(Fed. Cir. 2001). In the event the jury failed to conclude that Viola anticipated the

'906 patent, Microsoft argued that the invention was obvious in light of Viola.

SIBIA Neurosciences, Inc. v. Cadus Pharmaceutical Corp., 225 F.3d 1349, 1356

(Fed. Cir. 2000) ("In appropriate circumstances, a single prior art reference can

render a claim obvious.").

Based on a misunderstanding of patent law, the district court excluded

compelling evidence of anticipation. In addition, the court refused to submit

Microsoft's defenses to the jury based on the limited, but still sufficient, evidence

the court allowed Microsoft to present. As a result, the '906 patent, which was

allowed after the inventors concealed Viola from the PTO (*see supra* at 10-15),

was *enforced* without jury scrutiny of that same prior art, despite the jury's clear

interest in "deciding if Pei Wei was the first inventor and not Doyle." (*Supra* at

23.) A new trial is necessary.

A.   **The District Court Misconstrued Section 102 and Thus Wrongly
          Kept an Anticipating Version of the Viola Browser from the Jury.**

Wei's Viola browser included every limitation of the claimed

invention months before the '906 inventors allegedly conceived their invention.

Wei conceived of and reduced his invention to practice in late 1992. (A101155-56; A101197; A170005.) *Union Carbide Chems. & Plastics Tech. Corp. v. Shell Oil Co.*, 308 F.3d 1167, 1189 (Fed. Cir. 2002) (anticipation under § 102(g) can be established by proving prior reduction to practice or prior conception and reasonable diligence in reducing to practice). On May 7, 1993, using the DX34 code, Wei demonstrated and explained his browser to Sun without confidentiality restrictions. (A101162-63; A101176; A170007.) Wei continued to refine his invention and, when Sun asked for a copy of the Viola code, he provided Sun with the then-current version, as reflected in DX37. The changes Wei made to the DX34 code to produce the DX37 code were not intended to alter the relevant functionality of his browser. (A101163.)

Microsoft was prepared both to demonstrate for the jury, and to describe through testimony, that the May 7 (DX34) version of Viola includes all limitations of the claimed invention. (A101156-59.) The court kept all the evidence regarding DX34 from the jury — the code itself, testimony and demonstrations of its capabilities and the fact of its use, emails between Wei and others, *etc*. In its view, Wei had, as a matter of law, abandoned, suppressed, or concealed his invention, a finding that led the court to reject both Microsoft's section 102(g) and section 102(b) defenses. (A101187-89.) Even the court

31

recognized the harm these rulings caused Microsoft. (A101362.) The rulings

reflect fundamental legal errors.

1. The Erroneously-Excluded Evidence Establishes Wei's Prior Invention and That Wei Did Not Abandon, Suppress, or Conceal That Invention.

The district court's ruling reflects indifference to the proper analysis

for determining when prior art is abandoned, suppressed, or concealed under

section 102(g). This Court's "case law distinguishes between two types of

abandonment, suppression, or concealment. The first is implicated when an

inventor actively abandons, suppresses, or conceals his invention from the public."

Dow Chem., 267 F.3d at 1342. Active or "intentional" abandonment, suppression,

or concealment "refers to situations in which an inventor designedly, and with the

view of applying it indefinitely and exclusively for his own profit, withholds his

invention from the public." *Fujikawa v. Wattanasin*, 93 F.3d 1559, 1567 (Fed. Cir.

1996). A second type of "abandonment, suppression, or concealment may be

inferred based upon the prior inventor's unreasonable delay in making the

invention publicly known." *Dow Chem.*, 267 F.3d at 1342. The district court

ignored this legal framework. Had it abided it, the court could not have found

abandonment "as a matter of law."

Any suggestion of intentional abandonment, suppression, or

concealment fails because there is no evidence that Wei manipulated the period of

32

market exclusivity for his invention. Wei never sought market exclusivity and

never intended to profit from Viola. Instead, he adhered to the ethos that Web-

related developments should be openly shared, and he openly discussed his

invention and shared his browser code with interested members of the Web

community. (A101201.) A finding of abandonment, suppression, or concealment

of this sort has no evidentiary support. Neither does the record support a finding

based on the second kind of abandonment, suppression, or concealment:

"unreasonable delay" in making public the invention.

The court instead accepted plaintiffs' argument that an inventor can

abandon, suppress, or conceal a particular version or embodiment of an invention.

(A101187-88.) According to the court, Wei abandoned, suppressed, or concealed

the DX34 Viola code by changing the Viola code after his May 7, 1993,

demonstration, and never again distributing the specific DX34 code. (A101188.)

This ruling fundamentally misperceives what is the "invention" that can be

"abandoned, suppressed or concealed" for purposes of section 102(g).

a. Evidence of an Invention Cannot Be "Abandoned, Suppressed or Concealed."

A patent should be denied if the "*invention* was made in this country

by another inventor who had not abandoned, suppressed or concealed *it*." 35

U.S.C. § 102(g)(2) (emphasis added). Textually, only the "invention" must not be

"abandoned, suppressed or concealed." In this case, the invention, as defined by the '906 claims, is a browser that can parse a tag in a Web page which specifies the location of an object external to the Web page, uses information associated with the object to identify and locate an executable application, and automatically invokes that executable application to display, and allow interaction with, the object inside the Web page. The DX34 Viola code is one version of such a browser and is evidence corroborating Wei's prior invention. But DX34 is not the "invention" itself. The "invention" is not any particular code version. "An invention can exist … even though it may later be refined or improved." *New Railhead Mfg., L.L.C. v. Vermeer Mfg. Co.*, 298 F.3d 1290, 1297 (Fed. Cir. 2002). Creating improved versions of an embodiment might result in replacing the older *version*, but it is the opposite of "abandoning" the *invention*. Section 102(g) provides no basis for excluding a particular version from evidence simply because that version was superseded as the invention was refined.

This textual reading of section 102(g) is reflected in this Court's cases. In interference cases, the party who last conceived of the invention sometimes asserts that the other party abandoned, suppressed, or concealed the invention by unreasonably delaying disclosure of the invention after first reducing it to practice. But delay between the first reduction to practice and public disclosure can be excused if the inventor was continuing to refine, perfect, or

34

improve the invention. *Lutzker v. Plet*, 843 F.2d 1364, 1367 (Fed. Cir. 1988) ("An

inference of suppression or concealment may be overcome with evidence that the

reason for the delay was to perfect the invention."); *Young v. Dworkin*, 489 F.2d

1277, 1281 n.4 (CCPA 1974) (same); *Dewey v. Lawton*, 347 F.2d 629, 632 (CCPA

1965) ("testing and refinement" of an invention over the period of one and a half

years before public disclosure does not "evidence an intent to conceal" the

invention). "The law does not punish an inventor for attempting to perfect his

process before he gives it to the public. In fact, reasonable experimentation is

frequently encouraged." *Frey v. Wagner*, 87 F.2d 212, 215 (CCPA 1937).

A contrary rule, such as that adopted by the district court, would

deprive the public of the benefit of diligent efforts to produce the most useful

product. The dynamic technological environment in which Wei was working in

the early 1990s provides a good example. Wei was developing a browser that

could enable a higher level of interactivity on the Web. His invention would

engender the development of Web pages that would call upon that interactivity,

applications that could enable it, and incorporation of fast-changing server and

browser technology and protocols unrelated to the innovative features of his

browser. It is laudable, not condemnable, that Wei continued to work on his

browser so that it would most efficiently and effectively provide the innovative

features he had invented. The public is served by efforts to improve an invention,

especially when there has been no delay in order to advance the inventor's commercial interests. *See City of Elizabeth v. Am. Nicholson Pavement Co.*, 97 U.S. 126, 137 (1877).

The court excluded DX34 because it erroneously asked whether Wei abandoned, suppressed, or concealed the DX34 code, which was merely *evidence* of Wei's prior invention. It should have asked whether Wei abandoned, suppressed, or concealed his *invention* — the anticipating Viola browser evidenced by DX34. That question has only one answer: No.

b. Wei Did Not Abandon, Suppress, or Conceal His Prior Invention.

Wei produced a series of improved versions of the Viola code over the weeks and months following his demonstration to Sun, all of which retained the anticipating functionality. Wei testified that the May 31, 1993, Viola code he provided to Sun (DX37) met all the limitations of the '906 claims (A101163), that the alpha version he released to interested members of the Web community met all those limitations (A101165), and that the beta version he made publicly available also met all the limitations (A101168).[14] Each of these versions was publicly

---

[14] Promptly upon learning of Doyle's claim to have invented the functionality Wei had already incorporated into Viola, Wei initiated a series of email exchanges with Doyle regarding Viola's capabilities and the timeframe in which Wei had developed the critical functions. (A170078-88.) Wei's message is consistent and persistent: he first invented the functionality claimed in the '906 patent.

disclosed by posting on the Internet for interested members of the Web community to download. DX37 was disclosed in late May 1993, just 7 months after Wei first reduced his invention to practice. The alpha version was publicly disclosed in October 1993, just 11 months after Wei's first reduction to practice. And the beta version was disclosed just 16 months after Wei first reduced his invention to practice. (*See supra* at 5-8.) The district court never suggested, nor could it, that any of these disclosures was so delayed that Wei should be deemed to have abandoned, suppressed, or concealed his invention. And even the short time periods between releases of the various versions of Viola code were due to Wei's continued efforts to improve the browser (A101201), which, as noted above, counsels *against* a finding of abandonment.

Plaintiffs argued and the district court found that the DX37 Viola code does not anticipate. They were wrong,[15] but even if they were right, *DX34* is still evidence that Viola was a prior invention under section 102(g), and there is ample evidence that Wei never intended to remove the '906 functionality from Viola. (*See* A101164.) So even if Wei's changes had so altered the DX34 code that

---

[15] Wei did make changes to his Viola code after demonstrating the DX34 version to Sun, but as he explained: "the differences were primarily in the Lexical Analyzer part of the language.... It has something to do with the Viola language itself, … it has nothing to do with the interactive object capability." (A101163.) As discussed below, Microsoft presented sufficient evidence to permit a jury to conclude that the DX37 code included all the '906 limitations. (*Infra* at 42-47.)

DX37 no longer included every element of the claimed invention (which they did not), Wei was prepared to testify that those changes were "bugs" introduced by mistake and corrected by the time he posted his alpha release in October 1993. (A101162-65.) DX37 thus was not an essential part of Microsoft's defense. Microsoft could simply have relied on DX34 to show both prior conception and reduction to practice and the alpha (and/or beta) release to demonstrate that Wei publicly disclosed the invention and never abandoned, suppressed, or concealed it.

There is a striking disparity between how the court treated early versions of Wei's browser and plaintiffs' browser. Wei's early version (DX34) was deemed abandoned because the code was modified. Plaintiffs' early version (November 1993) — of which there is no existing code (A100725) — was treated as the reduction to practice even though it, too, was modified (into the code submitted with the patent, which does not work). (A35-36; A100725-26.) There is no basis in law or logic for this disparity.

In the end, when the section 102(g) case that Microsoft tried to present is properly analyzed, Microsoft's right to present the DX34 version of Viola code is indisputable. Microsoft should have been permitted to present DX34, and if plaintiffs had any evidence of abandonment, they could have presented it to the jury. Microsoft then should have been permitted to rely on evidence regarding

38

DX37, the alpha and beta releases, and various communications to demonstrate

that Wei consistently pursued his invention, improving it and sharing it with

interested members of the public, a showing corroborated by evidence that Wei

freely communicated about his invention almost as soon as he reduced it to

practice. (*E.g.*, A170004-06; A170011.) Finally, Microsoft should have been

permitted to further rebut any suggestion that Wei abandoned his invention with

evidence of Wei's quick response to Doyle's claims that Doyle had invented the

functionality that Viola had long included. (A170078-88.)

The jury, even without seeing Microsoft's best evidence, and even

after being told to disregard Viola, questioned whether Pei Wei, and not Doyle,

was the real inventor of the invention claimed in the '906 patent. (A101468.) This

Court should permit a jury to answer that question for itself, with full information.

2. A Jury Could Conclude That the Demonstration to Sun on May 7, 1993, Was a Prior Public Use.

Microsoft's argument that the May 7, 1993, demonstration of Viola to

Sun was an invalidating public use under section 102(b) provides an independent

ground for admitting the DX34 code, and testimony and documents regarding its

capabilities. "[P]ublic use … includes any use of the *claimed invention* by a

person other than the inventor who is under no limitation, restriction or obligation

of secrecy to the inventor." *Smithkline Beecham Corp. v. Apotex Corp.,* 365 F.3d

1306, 1317 (Fed. Cir. 2004). Oral testimony of prior public use must be corroborated. *Juicy Whip, Inc. v. Orange Bang, Inc.*, 292 F.3d 728, 737-38 (Fed. Cir. 2002). Microsoft presented direct and corroborating evidence of prior public use.

Wei's demonstration to Sun closely parallels the circumstances in *Netscape Communications*, 295 F.3d 1315. In *Netscape*, Konrad had three patents related to "systems that allow a computer user to access and search a database residing on a remote computer." *Id.* at 1318. The district court held Konrad's patents invalid based upon, *inter alia*, Konrad's prior demonstration of his invention to two "University of California computing personnel, without any obligation of confidentiality." *Id.* at 1319. This Court affirmed, holding that the "[l]ack of a confidentiality agreement is significant here because [the two individuals] were computer personnel who could easily demonstrate the invention to others." *Id*. at 1321. While an experimental use is not an invalidating public use, *id*. at 1320, Konrad's demonstration was not experimental because "the purpose of the demonstration 'was to convince the people in the Berkeley computer center … that there was a viable project.'" *Id*. at 1321-22.

Just as in *Netscape*, this case involves a demonstration of a software invention to interested and knowledgeable individuals in the field who were under

40

no confidentiality obligation. (A101163; A2326-28; A170007-10.) Indeed, the

Sun engineers sought out the demonstration precisely so they could share it with

others. (A170007 (Sun engineer "would like to see Viola because he's looking for

a few applications to show off this new architecture that he's working on.");

A170008-10 (Sun engineers "were very impressed with what we are doing, and

couldn't wait to show it around at Sun.").) Wei's demonstration was not

experimental. The whole point was to show Sun a working embodiment of the

browser that would generate interest in the software. (A2326-27.)

Despite compelling parallels with *Netscape*, the district court rejected

Microsoft's public use defense. The court held that because Wei later made

changes to the specific Viola code demonstrated to Sun, and never returned to that

precise code, the May 7, 1993, demonstration was not a public use. (A101188-89.)

When Microsoft pointed out that the court was conflating its erroneous view of

abandonment under section 102(g) with public use analysis under section 102(b),

the court stuck to its position: "If I'm wrong, I'm wrong." (A101192-93.)

The court was wrong. Nothing in *Netscape*, or any other decision,

suggests that a public use can somehow retroactively be made *un*public based on

subsequent actions. Nor would the policy underlying section 102(b)'s public use

bar be served by such a rule. Section 102(b)'s public use bar "encourages prompt

filing of patent applications after inventions have been completed and publicly used, and sets an outer limit to the term of exclusivity." *Allied Colloids, Inc. v. Am. Cyanamid Co.*, 64 F.3d 1570, 1574 (Fed. Cir. 1995). Once an invention has been exposed to members of the public, the "outer limit to the term of exclusivity" has been set. To allow later conduct to move that "outer limit" is directly at odds with the purpose of the public use bar.

Wei's May 7, 1993, demonstration was a public use of the claimed invention. It was error for the district court to exclude DX34 and the corroborating evidence that Microsoft proffered to support its public use defense.

B.   The Court Erred by Refusing To Submit Microsoft's Anticipation and Obviousness Defenses to the Jury.

Even after Microsoft's evidence was substantially pared down by the erroneous rulings, Microsoft still presented the jury with clear and convincing evidence that Viola anticipated the '906 patent. Denied the opportunity even to discuss the DX34 Viola code, Microsoft centered its defense on the DX37 code. Plaintiffs disputed DX37's teaching of the claimed invention, and therefore argued that DX37 cannot invalidate the patent under section 102(a), (b), or (g). The court agreed, stating cryptically that "DX37 is missing important elements and that those important elements are not taught by DX37" (A101362), without identifying which elements were missing and not taught.

42

Whether a prior art reference teaches every element of the claims is a question of fact. *Apple Computer, Inc. v. Articulate Sys., Inc.*, 234 F.3d 14, 20 (Fed. Cir. 2000). Therefore, JMOL regarding anticipation could be granted only if Microsoft had failed to present *any* evidence to support a finding that DX37 taught all limitations of the claimed invention. *Monsanto Co. v. Mycogen Plant Science, Inc.*, 261 F.3d 1356, 1362 (Fed. Cir. 2001).

Microsoft presented substantial evidence that the DX37 version of Viola taught every limitation. Microsoft's expert, Dr. Kelly, methodically explained this to the jury. (A101274-79.) Dr. Kelly testified that DX37: (1) is in a computer usable medium (A101277), (2) teaches a computer program for use in a distributed hypermedia environment (A101277), (3) teaches parsing a hypermedia document to identify text formats (A101277-78), (4) teaches displaying at least a portion of the hypermedia document on a client workstation (A101278), (5) teaches how the hypermedia document can include an embed text format that specifies the location of an object external to the hypermedia document (A101278), (6) teaches how the object can have type information associated with it that is utilized by the browser to identify and locate an executable application (A101278), and (7) teaches that the application is automatically invoked to display the object and enable the user to interact with it (A101278). Wei, too, explained

how DX37 includes each limitation of the claimed invention. (A101197-99; A101201.)

The court disregarded Dr. Kelly's detailed testimony as "conclusory." (A101354.) But Dr. Kelly did far more than merely state, as the court erroneously recalled, "I'm an expert, and it's there." (A101354.) He pointed to specific parts of the DX37 code that taught specific elements of the claimed invention. (A101275-77.) And he demonstrated DX37 for the jury. (A101282-83.) Though the court apparently did not think the jury was capable of determining which expert's testimony was more credible (A101354), that is precisely the jury's role. *Railroad Dynamics, Inc. v. A. Stucki Co.*, 727 F.2d 1506, 1514 (Fed. Cir. 1984).

It is true that the demonstration Dr. Kelly performed was not in a distributed hypermedia environment; the Web page was retrieved locally from the same computer on which the browser was running. (A101282.) The court thus apparently concluded that DX37 could not work in such an environment, and therefore did not anticipate. (A38 (stating that the code was not "enabled for use in a distributed hypermedia network environment").) But there was ample testimony that the DX37 code would work in such an environment.

Dr. Kelly testified that two problems can arise when using the DX37 code in a distributed hypermedia environment. First, DX37 will display a

hypermedia document retrieved from a remote server only if the server sends the document using the right HTTP protocol. (A101284.) As discussed above (*supra* at 21 n.9), this early Viola code had not been updated for the new protocol version, HTTP 1.0. But there was no dispute that it could display documents sent under the immediately-previous protocol, HTTP 0.9. (A101284; *see also* A101160.) Second, if the object is in a location remote from the browser, DX37 will not display the object. But if the object is local, DX37 will work in a distributed hypermedia environment without any modification. (A101284.) The '906 patent does not require the *object* to be retrieved from a remote source. The object must be external to the Web page, but can reside *anywhere* else, including the local computer as in Dr. Kelly's demonstration. (A150020, col. 17, lns. 7-9; A150020, col. 18, lns. 8-10; *see also* A101285.)

Thus, neither of these problems renders DX37 unable to work in *a* distributed hypermedia environment, which is all that the claims require. (A150019, col. 16, lns. 66-67; A150020, col. 17, ln. 61.) If the server sends documents using HTTP version 0.9, and the object is local, DX37 will function as the claims of the '906 patent require. The DX37 version of Viola can be prior art even if it does not work in "the circumstances of the invention's ultimate use." *Taskett v. Dentlinger*, 344 F.3d 1337, 1341 (Fed. Cir. 2003), *quoting Scott v.*

Finney, 34 F.3d 1058, 1063 (Fed. Cir. 1994). The jury should have been permitted to consider this evidence.

More fundamentally, "[e]ven if a reference discloses an inoperative device, it is prior art for all that it teaches." *Beckman Instruments, Inc. v. LKB Produkter AB*, 892 F.2d 1547, 1551 (Fed. Cir. 1989). As noted above, Dr. Kelly and Wei both testified that DX37 does *teach* all the elements of the '906 invention, regardless of whether it works.

Finally, notwithstanding plaintiffs' arguments, the '906 invention still could have been found obvious in light of Viola. The court held otherwise as a matter of law because it concluded that the evidence showed only how "someone could *post hoc* have figured it out." (A101362.) But that is *always* true of obviousness. If someone previously had actually done it, the defense would be anticipation. A patent is invalid as obvious when one of ordinary skill could have figured it out without inventive activity and had the incentive and capacity to do so. *B.F. Goodrich Co. v. Aircraft Braking Systems Corp.*, 72 F.3d 1577, 1582 (Fed. Cir. 1996).

The record supports such a determination. There would have been ample motivation to implement necessary changes if a user employed DX37 in a particular distributed hypermedia environment in which the browser did not work.

DX37 was a Web browser; its whole purpose was to work in such an environment. Even Doyle admitted that it would have been obvious in 1993 to use a browser in a distributed hypermedia environment. (A100721.) The changes, if any were necessary, were minor — a mere two lines of code. (A101283-84.) And whatever problems existed with DX37 were quickly resolved by Wei no later than his alpha release. (A101162-65.)

By granting JMOL as to anticipation and obviousness, the district court improperly substituted its judgment for that of the jury. Microsoft is entitled to a new trial.

III. THE DISTRICT COURT'S FAILURE TO APPLY THE APPROPRIATE LEGAL STANDARDS REQUIRES A NEW INEQUITABLE CONDUCT TRIAL.

Inequitable conduct based on a failure to disclose material information requires clear and convincing evidence of "(1) prior art or information that is material; (2) knowledge chargeable to applicant of that prior art or information and of its materiality; and (3) failure of the applicant to disclose the art or information resulting from an intent to mislead the PTO." *FMC Corp. v. Manitowoc Co.*, 835 F.2d 1411, 1415 (Fed. Cir. 1987).

It is undisputed that the '906 inventors failed to disclose Viola to the PTO. It is also undisputed that they had substantial information concerning Viola.

47

Doyle knew that Wei claimed to have first developed a browser with the capabilities claimed in the '906 patent. (A29-30 (describing emails between Wei and Doyle).) Doyle knew that others believed that Wei's Viola browser included those capabilities. (A29 (describing Raggett email).) In the midst of his first email exchange with Wei, Doyle even downloaded and read a paper on Viola. (A30; A101671-72; A101737-38.) Doyle, perhaps curious, concerned, or both, continued to investigate while his patent application was pending. Doyle reviewed a presentation Wei gave concerning Viola at Stanford University, which described the key feature of the '906 patent — embedding program objects in hypermedia documents — and illustrated precisely what Doyle claimed to have invented — interactive images displayed inside a browser window. (A33.) Doyle obtained descriptions of two beta versions of Viola. (A33.) As his knowledge of Viola's capabilities grew, Doyle suddenly stopped his inquiries. In particular, although he knew where to find Viola source code (A33), he chose not to get it (A101717-24), "cultivat[ing] ignorance [and] disregard[ing] numerous warnings that material information … may exist, merely to avoid actual knowledge of that information." *FMC Corp. v. Hennessey Indus.*, 836 F.2d 521, 526 & n.6 (Fed. Cir. 1987); *see Brasseler, U.S.A. I*, 267 F.3d at 1376. (A101717-18; A101721-22.)

Doyle plainly recognized that Viola was material. As soon as Wei explained that Viola had long had the capability that Doyle claimed to have

invented, Doyle inquired "[h]ow many months and months" earlier Viola had that capability. (A170079.) Wei told Doyle of the May 1993 demonstration. (A170082-83.) Doyle later mischaracterized Wei's statements, seeking an admission that Viola lacked the relevant capability until after Doyle had demonstrated it. (A170087.) Wei rejected Doyle's mischaracterization. (A170084.) As the court recognized, Doyle "sought to protect the value of the invention by committing Wei to an invention date," and gave false testimony at trial regarding his motive in doing so. (A39.)

Despite these facts, the district court refused to find that Doyle should be charged with knowledge of material information, a finding that would have virtually compelled a finding of intent. *Bristol-Myers Squibb Co. v. Rhone-Poulenc Rorer, Inc.*, 326 F.3d 1226, 1239 (Fed. Cir. 2003) ("[W]here withheld information is material and the patentee knew or should have known of that materiality, he or she can expect to have great difficulty in establishing subjective good faith sufficient to overcome an inference of intent to mislead."). The court did so because of the same fundamental misconception that infected its other Viola rulings: the court treated each version of the Viola code as a separate piece of prior art, wholly divorced from what preceded and what followed it, and regarded other information concerning Viola as irrelevant. Thus, the court focused on Viola code that predated Doyle's claimed date of conception (DX34 and DX37), and

49

because that particular code was, in the court's erroneous view, both "hard to obtain" (A34) and not invalidating (A34; A38), held that Doyle did not have material information. But the relevant prior art was the Viola browser, not a particular version of its code, and the wealth of information available to Doyle put him on notice that, at a minimum, Wei had conceived a browser with the '906 capabilities earlier than Doyle's claimed conception date and had reduced it to practice soon thereafter.

Knowledge of a potentially invalidating invention is plainly material. *See GFI, Inc. v. Franklin Corp.*, 265 F.3d 1268, 1274 (Fed. Cir. 2001) (applicant must disclose potential priority conflict and cannot unilaterally determine that device is not prior art). Doyle's decision to "cultivate ignorance" and avoid additional information concerning that invention does not diminish the materiality of what he already actually knew. Had the court applied the correct standard, it would have concluded that the information of which Doyle knew or should have known was "material," that he recognized its materiality, that he intentionally failed to disclose that information, and that, in light of Viola's materiality and Doyle's knowledge of it, the evidence compelled an "inference of intent to mislead." *Bristol-Myers Squibb Co.*, 326 F.3d at 1239. A new trial is necessary.

IV. THE DISTRICT COURT ERRONEOUSLY CONSTRUED THE CLAIMS.

A. "Executable Application" Does Not Include Components Such as DLLs.

The '906 patent requires that an "executable application" be automatically invoked to process an embedded data object. (A63.) The district court instructed the jury that "executable application" is "any computer program code that is not the operating system or a utility" and that it may include "a component such as a dynamic link library [DLL], or multiple components working together." (A101462-63.) This construction is inconsistent with the ordinary meaning of the claim language, the specification, and the prosecution history.

The ordinary meaning of "application" is "[a] *computer program designed to help people perform a certain type of work.*" *Microsoft Press Computer Dictionary*, 2d ed., 1994, pp. 23-24.[16] A "computer program" is:

> A set of instructions in some computer language, intended to be executed on a computer to perform a useful task. *The term usually implies a self-contained entity, as opposed to a routine or a library.*

---

[16] The '906 applicants relied upon the Second Edition of the *Microsoft Press Computer Dictionary* during prosecution. (*See* A150913.) "[D]ictionaries, encyclopedias and treatises are particularly useful resources" for construing claim terms. *Texas Digital*, 308 F.3d at 1202.

Id. at 90 (emphasis added). Thus, "application" "usually" refers to "a self-contained entity," not "a routine or a library." That this "usual" meaning applies here is confirmed by the claims' use of "executable." The ordinary meaning of "executable," when used with "application" or "program," requires that the application or program is ready to run: "**executable program:** A computer program that is *ready to run . . . .*" *Id.* at 153 (emphasis added). A computer program that is not ready to run is not an "executable application."

As construed below, "executable application" covers applications that are not "executable" — *i.e.*, components, such as routines, libraries, and DLLs, which are not ready to run because they require that another application be executed and running first so that they can run within it. But the claims require "*executable* applications." This Court has made clear that claim scope cannot be expanded by reading language out of a claim. *See, e.g., Apple Computer*, 234 F.3d at 24-25 (rejecting construction of "help access window" which encompassed a window that accesses any information, rather than just "help" information); *Altiris, Inc. v. Symantec Corp.*, 318 F.3d 1363, 1372 (Fed. Cir. 2003) (each word in a claim phrase should be given its established meaning). The only construction of this limitation that is true to the ordinary meaning of each claim term construes "executable application" as a standalone application that can be independently executed by the user.

The district court identified no explicit definitions in the specification

that deviated from the terms' ordinary meanings. Instead, the court discarded the

ordinary meanings based on a "logical inference" from the specification's

discussion of another subject altogether, the use of modules to construct the

browser code. (A65-67.) But even if the court's "inference" were "logical," and it

was not, it is not the explicit definition needed to rebut the presumption that

ordinary meaning applies. *Texas Digital*, 308 F.3d at 1204. Further, *every*

disclosed embodiment describes a standalone program that can be independently

run by a user. (*See, e.g.,* A150016, col. 10, lns. 17-27; A150018, col. 13, lns. 11-

15; A150015, col. 7, lns. 29-30; A150017, col. 11, lns. 40-41; *see also* A69.)[17]

While a claim is not limited to the disclosed embodiments, *Northern Telecom Ltd.*

v. Samsung Electronics Co., 215 F.3d 1281, 1293 (Fed. Cir. 2000), disclosed

embodiments consistent *only* with the ordinary meaning of claim language make

clear that the ordinary meaning should apply. *See Renishaw PLC v. Marposs*

Societá per Azioni, 158 F.3d 1243, 1250-53 (Fed. Cir. 1998).

The prosecution history powerfully supports construing the claims to

cover standalone applications and not components such as DLLs. (*See supra* at

---

[17] Likewise, the code the applicants submitted with their application was written
for use only with standalone applications, not DLLs or other components.
(A101272.)

12-15.) The applicants argued that Khoyi used libraries of routines (*i.e.*, components), while their invention did not. (*See* A150344.) Similarly, they argued that Koppolu taught away from using standalone applications — *i.e.*, server applications that run "'as a separate process from the container application'" — and instead disclosed using "special code" that "provides a subset of the functionality of the full server application." (A150905.) Specifically, Koppolu used DLLs — "object handlers" and "in-process servers" — to provide the claimed functionality. (A150905-07.) The applicants went on to stress the disadvantages of DLLs generally and to tout the advantages of their invention — which used "executable applications" — over the prior art use of DLLs. (A150906-07.) The Examiner expressly relied on these representations: "The examiner agrees that the claimed external executable application is not a code library extension nor object handler (e.g. windows dll and OLE) as pointed out in applicant's argument (paper #19 pages 12-14)." (A151031.)

The court asserted that the Examiner was referring only to particular varieties of "windows dll" discussed in "Paper #19" — in-process servers and object handlers — and not to DLLs or components as a class.[18] (A79.) But the

---

[18] The district court ultimately failed to give the Examiner's statement even this limited effect. The court's construction of "executable application" (*see* A101462-63) excludes *nothing*, not even the in-process servers and object handlers on which the Examiner supposedly focused.

applicants, in an interview with the Examiner, had already tried to distinguish

Koppolu on the grounds that OLE's particular DLLs were different, arguing that

OLE taught automatic invocation of applications that did not allow for

interactivity. (A150521.) Only after that effort failed did they file Paper #19, in

which they distinguished the prior art more broadly, arguing that DLLs by their

very nature are different — *i.e.*, they do not run standalone. They thus focused on

their invention's ability to have an "entire server application" automatically run

within the embedded window, in contrast to Koppolu's automatically-invoked

DLLs. (A150901.) This effort to distinguish Koppolu paralleled the broad

argument advanced to distinguish Khoyi: the general category of "libraries of

routines" was used by the prior art, but not by their invention. (A150344.) It was

in this context, and with that understanding, that the Examiner allowed the patent.

It is undisputed that ActiveX controls, applets, and plug-ins are

components and not standalone applications. (*See*, *e.g.*, A101069; A101116;

A101120.) This Court — applying a proper construction of the claims — should

therefore hold as a matter of law that the accused products do not infringe. *See*

Electro Scientific Indus., Inc. v. Dynamic Details, Inc., 307 F.3d 1343, 1350

(Fed. Cir. 2002) (JMOL may be entered on appeal if, under correct claim

construction, no genuine issues of material fact exist). Alternatively, a new trial

should be granted because a properly-instructed jury could have found for

Microsoft. *See Ecolab, Inc. v. Paraclipse, Inc.*, 285 F.3d 1362, 1373 (Fed. Cir.

2002).

B.   The Claims Require That the Executable Application Be Identified and Located by the Browser, Not the Operating System.

The '906 claims require that "type information" be "utilized by [the]

browser to identify and locate an executable application." (A150020, col. 17,

lns. 17-19.) The inventors relied on this feature to distinguish Khoyi and Koppolu

(the OLE prior art) and obtain their patent. OLE is a Microsoft technology that,

like the '906 invention, allows an object to be displayed by one application within

the window of another application. (*See supra* at 12-14.) OLE, which can work

with any application, utilizes type information to identify and locate the executable

application that will display the object. The "type information" used by OLE is a

"CLASS_ID." (A150459, col. 10, lns. 7-10 ("The class identifier (CLASS_ID) is

used to access the appropriate server application for the object. It is similar to a

data structure "type" used in programming languages."); A101082.) Critically,

though, in OLE the *operating system* uses the "type information" (the CLASS_ID)

to make the ultimate determination of which application to launch — *e.g.*, it

consults its own registry to determine which application should display the object

(thereby identifying the application), and determines where the relevant code can

be found (thereby locating the application). (A150460, col. 11, lns. 6-15.)

During prosecution, the inventors distinguished this operating-system-dependent OLE prior art precisely because the operating system performs the identifying and locating functions. (A150342; A150344.) The inventors were clear: "OLE references the operating system's global registry database in order to identify which server application is related to a particular data object." (A150902.) Their invention was different, they claimed, because the browser, not the operating system, uses the type information to identify and locate an executable application, supposedly allowing the browser to function independent of any particular operating system.

OLE thus provides the outer limit of the '906 claims. If a browser, after parsing "type information" in a Web page, relies on the operating system to "identify and locate" an executable application — *i.e.*, to determine which application to run and where the relevant code can be found — the browser does not infringe. The district court agreed in its *Markman* ruling: "the functions of utilizing the type information to identify and locate the executable application must be performed by the browser, not the operating system as in Koppolu's OLE." (A88.) In short, the inventors *dis*claimed a program that uses the operating system to identify and locate the executable application. (A87-88.)

Microsoft demonstrated at trial that the accused products fit the inventors' disclaimer. To be sure, IE "utilizes type information." The critical question is: what for? Plaintiffs' expert was clear: IE uses the "type information" in a Web page to derive a binary CLASSID. (A100869 (plaintiffs' expert admitting that the product of the relevant IE processes is a binary CLASSID).) The binary CLASSID is then passed to the operating system, which uses that information to "identify and locate" an executable application, *just as in OLE*. Indeed, the ActiveX controls ultimately located and invoked by the operating system are OLE controls. (A101069; A101071-72; A101082.) The operating system maintains the relevant registry database, compares the CLASSID to the registry, determines which application to run and where the files are located, and launches *that* specific application. (A100867-69; *cf.* A100841; A101260.) On this record, a properly-instructed jury could not have found infringement.

The jury, however, was not properly instructed. Although the court told the jury that "the enumerated functions," identify and locate, "are performed by the browser," it went on to instruct that "[t]he inventors contemplated the browser's use of some outside resources such as the operating system…." (A101463.) There is no reason to discuss the role of the operating system in this context except to convey the impression that the claims encompass "the browser's use of … the operating system" to identify and locate the application — *i.e.,*

58

precisely what was disclaimed by the inventors to distinguish OLE. At a minimum, the instruction as given, without additional guidance, introduced enough ambiguity that the jury could have read it to encompass the very OLE technology distinguished to obtain the patent. Microsoft, which had proved that IE (by passing along a binary CLASSID) relies upon the Windows operating system to identify and locate the application, urged the court to be clear. Microsoft asked for an instruction consistent with the *Markman* ruling: "[T]he claim does not apply where the computer's operating system, such as by consulting the operating system registry, determines what application will be invoked and where the executable files for that application are located." (A1800.) To the same end, Microsoft also proposed that the court define "identify" and "locate." (A1778-80; A1799 ("identify" means to determine "what executable application to launch," and "locate" means to determine "where the computer code for that executable application can be found on the user's computer.").)

The court rejected these proposals and neither instructed the jury concerning the boundary on claim scope that OLE necessarily represented, nor gave the jury the definitions of key claim terms that would have led it to find that boundary on its own. This was error. *See Moba, B.V. v. Diamond Automation, Inc.*, 325 F.3d 1306, 1313 (Fed. Cir. 2003) ("Claim construction is a question of law and is not the province of the jury.").

Armed with the erroneous instruction, plaintiffs' counsel encouraged

the jury to find infringement even though in Microsoft's accused products, just as

in Koppolu's OLE, the operating system consults its own registry to determine

what application to invoke (file name) and where the relevant code is located (file

path). (A101457 ("nowhere in [the] instruction will it say that identify means file

name or that locate means file path").) Indeed, seizing on the instruction's

infirmity, plaintiffs' counsel identified as a fact in plaintiffs' favor that Windows

does for IE just what it does for Word and for Excel, even though what Windows

does for Word and Excel is the core of the OLE prior art that the inventors

distinguished. (A101433; *see* A101297.) The jury instruction abandoned the

limits established by the disclaimer in light of OLE, and the jury accordingly gave

no effect to those limits. Because a properly-instructed jury would have been

compelled to find that the accused products do not infringe, judgment should be

entered for Microsoft. *See Electro Scientific Indus.*, 307 F.3d at 1350.

Alternatively, a new trial is required. *See Ecolab,* 285 F.3d at 1374, 1376.

V.   FOREIGN SALES OF UNITS MADE OUTSIDE THE UNITED STATES DO NOT INFRINGE UNDER SECTION 271(f).

Section 271(f) creates a limited exception to the general rule that U.S.

patents do not have extraterritorial effect, by imposing liability on those who

supply "*components*" of a patented product from the United States in "such manner

60

as to actively induce the *combination of such components* outside of the United States." 35 U.S.C. § 271(f)(1) (emphasis added); *see also* 35 U.S.C. § 271(f)(2). At issue here is whether, when Microsoft sends "golden master" disks to foreign OEMs, it supplies "components" of a patented invention for "combination" abroad. The court interpreted section 271(f) to include as an act of infringement the use of *information*, not parts, sent from the U.S. when making a foreign-made product. This was error.

Section 271(f) was enacted in response to a perceived "loophole" created by *Deepsouth Packing Co. v. Laitram Corp.*, 406 U.S. 518 (1972). 130 Cong. Rec. H. 10525 (Oct. 1, 1984), 98th Cong., 2d Sess., *reprinted in* 1984 U.S.C.C.A.N. 5827, 5828. In *Deepsouth*, the Supreme Court held that manufacturing the components of a patented shrimp deveining machine in the U.S. but assembling those components abroad did not infringe a patent that would have been infringed if assembly had taken place in the U.S. *Deepsouth*, 406 U.S. at 523-26. In *Deepsouth*, the components made in the U.S. were combined into the final product abroad, and each U.S.-made component became a part of an individual foreign-made unit. Section 271(f) prevents evasion of a patent by the expedient of completing the manufacturing process — assembly — abroad.

Providing foreign OEMs with "golden masters" has nothing to do with the loophole section 271(f) closes. A golden master is never itself physically incorporated into the final product. Only intangible information from the golden master is copied into the foreign-made and sold computer. The "golden master" is thus like a prototype or mold or detailed set of instructions used to make products abroad, but which never itself becomes a component of a foreign-made product. If, for example, a single prototype tire were shipped abroad and copied to make tires for sale abroad, section 271(f) would not be implicated. The prototype never becomes a part, or "component," of the foreign tires. Rather, the foreign-made products reflect only information from the prototype — its size, tread pattern, *etc*.

Similarly, when a copy of Windows is made abroad using a "golden master," the "golden master" continues as a separate entity, unchanged, just as products made using a prototype or mold do not alter the prototype or mold. No part of the "golden master" is physically incorporated into a foreign-made product, just as no part of a prototype or mold is a component of products made using them. Thus, Microsoft's act of shipping the "golden master" abroad does not involve or induce the "combination" of U.S.-made "components," as required under section 271(f).

Applying section 271(f) to foreign-made copies of computer software

leads to startling results, far removed from Congress's intent, or even power.

Under that interpretation of the statute, computer-related products made and sold

anywhere in the world would be subject to the U.S. patent laws, if they use

software code originally developed in the U.S., even if all physical parts of those

products were made and assembled abroad. Indeed, under this view, any patented

product that is made abroad using information obtained from U.S.-made

prototypes, blueprints, or molds would fall within section 271(f)'s reach.

Section 271(f) would thus give the patent laws sweeping extraterritorial effect,

contrary to settled principles. *See Deepsouth*, 406 U.S. at 531 ("Our patent system

makes no claim to extraterritorial effect."); *Dowagiac Mfg. Co. v. Minnesota

Moline Plow Co.*, 235 U.S. 641, 650 (1915) ("The right conferred by a patent

under our law is confined to the United States and its territories."). There is no

indication — either in the statutory language or legislative history — that Congress

intended such a revolution. This Court has indicated that such an extension of the

patent laws is exclusively a matter for Congress, not the courts. *See Bayer AG v.

Housey Pharmaceuticals, Inc.*, 340 F.3d 1367, 1376-77 (Fed. Cir. 2003) ("[W]e

think it is best to leave to Congress the task of expanding the statute if we are

wrong in our interpretation" because "Congress is in a far better position to draw

the lines that must be drawn if the product of intellectual processes rather than manufacturing processes are to be included within the statute.").

Finally, this Court has already interpreted the word "component" in section 271(f)'s companion provision, section 271(g). In *Bayer*, this Court concluded that the phrase "trivial and nonessential component" in section 271(g)(2) contemplates "*a physical product*," not information. 340 F.3d at 1373 (emphasis added). "'[I]dentical words used in different parts of the same act are intended to have the same meaning.'" *Gustafson v. Alloyd Co., Inc.,* 513 U.S. 561, 570 (1995). There is no reason to expansively construe "component" in section 271(f) to include intangible information, but narrowly construe it in section 271(g)(2) to exclude it.

When section 271(f) is properly construed, non-U.S. sales of Windows with IE must be excluded from the jury's award, which requires a 64 percent reduction.[19] (*See* A2320 (during the relevant time period, 226,639,360

---

[19] Plaintiffs argued that exclusion of foreign sales would result in a 42.4 percent reduction, but they based their calculation on data that attribute sales based on the *billing address* of the customer, not the site where the units are actually *made*. (*See* A2771.) The proper question is how many units were made in the U.S. and how many were made abroad. Plaintiffs have not disputed the accuracy of the data underlying Microsoft's 64 percent figure.

units produced abroad while 127,484,640 units domestically produced).) Thus, even if the jury's verdict is otherwise upheld, the award should be reduced from $520,562,280 to $187,402,420, with a commensurate reduction in prejudgment interest.

CONCLUSION

The judgment of infringement should be reversed or, in the alternative, vacated and remanded for new trial under a proper claim construction. The judgment holding the patent not invalid or unenforceable should be vacated

and remanded for new trial. Alternatively, the damages award based on foreign

sales should be reversed.

Respectfully submitted,

_____

Carter G. Phillips
Sidley Austin Brown & Wood LLP
1501 K Street, N.W.
Washington, D.C. 20005
202-736-8000

David T. Pritikin
Constantine L. Trela, Jr.
Richard A. Cederoth
Robert N. Hochman
Sidley Austin Brown & Wood LLP
Bank One Plaza
10 South Dearborn Street
Chicago, IL 60603
312-853-7000

Of Counsel: H. Michael Hartmann
                               Brett A. Hesterberg
T. Andrew Culbert   Steven P. Petersen
Microsoft Corporation   Leydig, Voit & Mayer, Ltd.
One Microsoft Way   Two Prudential Plaza
Building 8, N/203   180 North Stetson Avenue, #4900
Redmond, WA 98052   Chicago, IL 60601-6780
425-882-8080   312-616-5600

Attorneys for Microsoft Corporation

CERTIFICATE OF COMPLIANCE WITH
FED. R. APP. P. 32(a)(7)(B)

Counsel for Microsoft Corporation certifies that the body of this brief,

beginning with the "Jurisdictional Statement" on page 1 and ending with the last

line of the "Conclusion" on page 66, contains 13,950 words, as measured by the

word-processing system used to prepare this brief, in compliance with

Rule 32(a)(7)(B)(i) of the Federal Rules of Appellate Procedure.

_____
Constantine L. Trela, Jr.
Counsel for Microsoft Corporation

CERTIFICATE OF SERVICE

I hereby certify that on the 3rd day of June 2004, two copies of the foregoing Brief of Defendant-Appellant Microsoft Corporation were served upon:

> Martin R. Lueck
> Robins, Kaplan, Miller & Ciresi
> 2800 LaSalle Plaza
> 800 LaSalle Avenue
> Minneapolis, MN 55402-2015

by delivery of said copies, properly addressed with shipping charges prepaid, to a commercial overnight delivery service with instructions to deliver said copies to the addressee the next business day.

I further certify that on the 3rd day of June 2004, the original and eleven copies of the foregoing Brief of Defendant-Appellant Microsoft Corporation were filed with the Court as follows:

> Clerk
> U.S. Court of Appeals for the Federal Circuit
> 717 Madison Place, N.W.
> Washington, DC 20439

by delivery of said copies, properly addressed with shipping charges prepaid, to a commercial overnight delivery service with instructions to deliver said copies to the addressee the next business day.

_____
Constantine L. Trela, Jr.