

# Exhibit R



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

---

REPLY 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

## TABLE OF CONTENTS

|  |    |
|--|----|
| TABLE OF AUTHORITIES.....  | ii |
| INTRODUCTION.....  | 1  |
| ARGUMENT.....  | 3  |
| I. THE DISTRICT COURT ERRONEOUSLY KEPT MICROSOFT’S<br>PRIOR ART DEFENSES FROM THE JURY.....                            | 3  |
| A. The Exclusion Of The DX34 Version Of Viola Was Error. ....  | 3  |
| 1. There Was Sufficient Evidence That the DX34 Version<br>of Viola Teaches Every Element of the Claims. ....           | 5  |
| 2. DX34 Was Not Abandoned.....   | 9  |
| 3. DX34 Was Independently Admissible To Establish<br>Microsoft’s Section 102(b) Prior Public Use Defense.....          | 13 |
| B. The Jury Should Have Been Permitted To Consider The Viola<br>Evidence That Was Admitted.....                        | 15 |
| II. A NEW TRIAL ON INEQUITABLE CONDUCT IS REQUIRED. ....   | 17 |
| III. THE DISTRICT COURT’S CLAIM CONSTRUCTION WAS<br>WRONG AS A MATTER OF LAW.....                                      | 19 |
| A. The Court Construed “Executable Application” Contrary To Its<br>Plain Meaning And The Intrinsic Evidence. ....      | 19 |
| B. The District Court Erred In Construing The “By the Browser”<br>Limitation.....                                      | 26 |
| IV. THE DISTRICT COURT ERRED AS A MATTER OF LAW BY<br>HOLDING THAT FOREIGN SALES INFRINGE UNDER<br>SECTION 271(f)..... | 30 |
| CONCLUSION .....   | 32 |

## TABLE OF AUTHORITIES

### CASES

|  |        |
|--|--------|
| <i>Apple Computer, Inc. v. Articulate Sys., Inc.</i> ,<br>234 F.3d 14 (Fed. Cir. 2000).....              | 5      |
| <i>Bayer AG v. Housey Pharms., Inc.</i> ,<br>340 F.3d 1367 (Fed. Cir. 2003).....                         | 31, 32 |
| <i>Bristol-Myers Squibb Co. v. Rhone-Poulenc Rorer, Inc.</i> ,<br>326 F.3d 1226 (Fed. Cir. 2003).....    | 18     |
| <i>Deepsouth Packing Co. v. Laitram Corp.</i> ,<br>406 U.S. 518 (1972).....                              | 31     |
| <i>Dow Chemical Co. v. Astro-Valcour, Inc.</i> ,<br>267 F.3d 1334 (Fed. Cir. 2001).....                  | 9      |
| <i>EMI Group N.A., Inc. v. Cypress Semiconductor Corp.</i> ,<br>268 F.3d 1342 (Fed. Cir. 2001).....      | 5      |
| <i>Ecolab, Inc. v. Paraclipse, Inc.</i> ,<br>285 F.3d 1362 (Fed. Cir. 2002).....                         | 26, 29 |
| <i>Electro Scientific Indus., Inc. v. Dynamic Details, Inc.</i> ,<br>307 F.3d 1343 (Fed. Cir. 2002)..... | 26, 29 |
| <i>FMC Corp. v. Hennessy Indus.</i> ,<br>836 F.2d 521 (Fed. Cir. 1987).....                              | 18     |
| <i>Fromson v. Anitec Printing Plates, Inc.</i> ,<br>132 F.3d 1437 (Fed. Cir. 1997).....                  | 23     |
| <i>Laitram Corp. v. Morehouse Indus., Inc.</i> ,<br>143 F.3d 1456 (Fed. Cir. 1998).....                  | 22     |
| <i>Multiform Desiccants, Inc. v. Medzam, Ltd.</i> ,<br>133 F.3d 1473 (Fed. Cir. 1998).....               | 25     |

|  |            |
|--|------------|
| <i>Netscape Communications Corp. v. Konrad</i> ,<br>295 F.3d 1315 (Fed. Cir. 2002).....                    | 13, 14, 15 |
| <i>Pellegrini v. Analog Devices, Inc.</i> ,<br>375 F.3d 1113 (Fed. Cir. 2004).....                         | 2, 30-32   |
| <i>Phillips v. AWH Corp.</i> ,<br>2004 WL 1627271 (Fed. Cir. July 21, 2004).....                           | 21         |
| <i>Plant Genetic Sys., N.V. v. DeKalb Genetics Corp.</i> ,<br>315 F.3d 1335 (Fed. Cir. 2003).....          | 21         |
| <i>Sandt Technology, Ltd. v. Resco Metal &amp; Plastics Corp.</i> ,<br>264 F.3d 1344 (Fed. Cir. 2001)..... | 12         |
| <i>Singh v. Brake</i> ,<br>222 F.3d 1362 (Fed. Cir. 2000).....   | 11         |
| <i>Texas Digital Sys. Inc. v. Telegenix, Inc.</i> ,<br>308 F.3d 1193 (Fed. Cir. 2002).....                 | 19         |
| <i>Texas Instruments Inc. v. United States Int'l Trade Comm'n</i> ,<br>988 F.2d 1165 (Fed. Cir. 1993)..... | 20         |
| <i>Toro Co. v. White Consolidated Indus., Inc.</i> ,<br>199 F.3d 1295 (Fed. Cir. 1999).....                | 22         |
| <i>W.L. Gore &amp; Associates, Inc. v. Garlock, Inc.</i> ,<br>721 F.2d 1540 (Fed. Cir. 1983).....          | 14, 15     |
| <i>Wright Medical Tech., Inc. v. Osteonics Corp.</i> ,<br>122 F.3d 1440 (Fed. Cir. 1997).....              | 20         |

**STATUTES**

|                         |           |
|-------------------------|-----------|
| 35 U.S.C. § 102(b)..... | 4, 13, 14 |
| 35 U.S.C. § 102(g)..... | 4, 9, 12  |

35 U.S.C. § 271(f) .....3, 30-32

**OTHER AUTHORITIES**

*IBM Dictionary of Computing*, 10th ed., 1994..... 21

*Microsoft Press Computer Dictionary*, 2d ed., 1994 ..... 19

## INTRODUCTION

Plaintiffs defend the district court's rulings, but not its reasoning.

Instead, relying on their own manufactured arguments, plaintiffs seek to salvage a judgment that is based on fundamentally erroneous analyses. Plaintiffs' after-the-fact alternatives to the district court's flawed reasoning cannot withstand scrutiny.

With respect to the extraordinary rulings that kept the Viola prior art from the jury, the district court's analysis is nowhere mentioned in plaintiffs' brief. Even more striking, plaintiffs' own arguments in support of those rulings are presented as if a jury had considered and rejected Microsoft's prior art defenses, and all plaintiffs need show is that a reasonable jury could have found for them. But cherry-picking evidence from the record cannot save this judgment. The district court refused to allow the jury to consider the invalidating Viola prior art. Unless the record as a whole, including evidence favorable to Microsoft, would permit no reasonable jury to find the patent invalid, a new trial is necessary. Plaintiffs refuse to confront this standard, preferring instead to ignore the voluminous evidence establishing that Viola invalidates the '906 patent.

The district court's failure to appreciate the significance of Viola undermines its inequitable conduct findings as well. Because the court believed that Viola was not material prior art, it dismissed Doyle's cultivated ignorance and



intentional failure to disclose what he knew. A proper understanding of Viola requires a different result.

Plaintiffs' claim construction arguments similarly fail to address the district court's errors. The court's construction of the "executable application" limitation violates a fundamental tenet of claim construction by failing to give effect to every word in the claims. It also ignores the prosecution history by giving the claims a scope that encompasses what the inventors and the Examiner alike recognized to be prior art. As for the "by the browser" limitation, the court at the *Markman* stage had correctly excluded from the scope of the claims a browser that relied on "the operating system as in Koppolu's OLE" to identify and locate the executable application. Then the court wrote that limitation out of its jury instruction. The effect: what had been Microsoft's shield against the infringement claim — that IE uses Windows the way OLE used its operating system — was turned into plaintiffs' sword. Plaintiffs mischaracterize Microsoft's argument rather than confront it.

Finally, this Court's decision in *Pellegrini v. Analog Devices, Inc.*, 375 F.3d 1113 (Fed. Cir. 2004), compels the conclusion that foreign-made products can infringe a U.S. patent only where components of the products were physically present in the U.S. and were exported for assembly abroad. Here, the

“golden master” disks — the only items physically present in and sent from the U.S. — do not become part of any foreign-made products. Section 271(f) cannot apply.

## **ARGUMENT**

### **I. THE DISTRICT COURT ERRONEOUSLY KEPT MICROSOFT’S PRIOR ART DEFENSES FROM THE JURY.**

Plaintiffs and Microsoft apparently agree on one important point: the district court offered indefensible reasons for excluding the DX34 version of Viola (A170022) and for entering JMOL on Microsoft’s anticipation and obviousness defenses. Plaintiffs flee the district court’s analysis, instead offering their own. But none of plaintiffs’ efforts to make sense of the prior art rulings justifies keeping compelling evidence of prior art — evidence that the ’906 inventors never submitted to the PTO — from the jury.

#### **A. The Exclusion Of The DX34 Version Of Viola Was Error.**

The district court ruled that the DX34 version of Viola was abandoned for no other reason than that Wei refined Viola after creating DX34, producing improved versions and never returning to the precise version reflected in DX34. (A101187-88.) The court never found that Wei’s changes to Viola made the later versions different from DX34 in any material way, namely in any way relevant to whether Viola anticipates the ’906 patent. According to the district court,

“abandonment, concealment, and suppression can apply to a version” of software, without regard for how earlier versions differ from later ones. (A101187.) Further, the court never explained, nor could it, why such “abandonment” would matter with respect to Microsoft’s section 102(b) public use defense.

Microsoft explained why this ruling is inconsistent with the text of section 102(g), this Court’s caselaw, and the policy behind the abandonment rule. (Microsoft Br. 33-36.) Modifying versions of software or any other invention in an effort to improve it is the very opposite of abandonment. And, Microsoft also explained that abandonment has nothing to do with public use. (Microsoft Br. 41.) Plaintiffs never confront Microsoft’s arguments.

Instead, plaintiffs first assert that DX34 does not anticipate because it does not teach all elements of the ’906 claims. (Pl. Br. 27-28.) Second, plaintiffs argue that Wei changed the “functionality of the VOBJF architecture as it existed in DX34” in an unspecified way and never returned to it (Pl. Br. 33), and that Wei abandoned the “VOBJF architecture” when he switched “to an architecture based on a <LINK> tag which had different capabilities” (Pl. Br. 34).

These arguments are wrong and factually unsupported. Plaintiffs simply ignore that the DX34 version of Viola was demonstrated in court in a hypermedia environment (A101158-60), which led even the district court to

recognize that DX34 presented a “much stronger” case for anticipation than Microsoft’s other evidence. (A101362.) Not only have plaintiffs run from the district court’s analysis, but they have retreated to an analysis that even the court found unsupportable. There was no good reason to keep this critical evidence of invalidating prior art from the jury.

**1. There Was Sufficient Evidence That the DX34 Version of Viola Teaches Every Element of the Claims.**

Because anticipation is a fact question to be decided by the jury, *Apple Computer, Inc. v. Articulate Sys., Inc.*, 234 F.3d 14, 20 (Fed. Cir. 2000), it is not enough for plaintiffs to find *some* evidence that might support a jury finding that the DX34 (or DX37) version of Viola fails to teach a claim element. Instead, plaintiffs must show that no reasonable jury could conclude that the version anticipates. *EMI Group N.A., Inc. v. Cypress Semiconductor Corp.*, 268 F.3d 1342, 1351 (Fed. Cir. 2001). Plaintiffs nowhere acknowledge either this burden or the substantial evidence of anticipation Microsoft presented. The evidence upon which plaintiffs rely to argue that Viola does not teach three elements of the claims is often irrelevant and always controverted. The jury should have been permitted to consider DX34.

a. “*A distributed hypermedia environment.*” First, plaintiffs claim that DX34 is not “a computer program product for use in a distributed hypermedia

network environment.” (Pl. Br. 27; *see also id.* at 19-20.) This unsupported assertion flies in the face of Wei’s testimony that he wrote the DX34 code “with the Internet in mind” (A101160); indeed, Viola is a World Wide Web browser and was called “ViolaWWW.” (*E.g.*, A101154; A170000.) Plaintiffs’ assertion also disregards Microsoft’s demonstration of the DX34 version of Viola in a distributed hypermedia network environment during *voir dire* of Wei. (A101158-60.) Had the court admitted the evidence, the demonstration would have been performed before the jury and would have been supported by testimony explaining how this version of Viola performed all the limitations of the ’906 claims, including operating in a distributed hypermedia network environment. (A2327-28; A2341-43.)

Plaintiffs argue that the DX34 version of Viola does not anticipate because: (1) Microsoft placed the hypermedia document and object on a server to demonstrate how Viola works in a distributed hypermedia environment (Pl. Br. 20); (2) the in-court demonstration over the Internet involved a server programmed to communicate in an HTTP protocol that this old version of Viola could process (A101160; Microsoft Br. 21 n.9); and (3) Wei testified that the public *demonstrations* he conducted of the DX34 code were not performed in a distributed hypermedia environment (A101176; A101178). All of these are irrelevant distractions.

The claims require the browser to work in “a distributed hypermedia environment” (A150019, col. 16, lns. 66-67; A150020, col. 17, ln. 61 (emphasis added)), and plaintiffs do not deny that Microsoft’s demonstration was in “a distributed hypermedia environment.” Whether Wei demonstrated DX34 to Sun in May 1993 in a distributed hypermedia environment says *nothing* about whether it had the capability to function in such an environment at the time, which is all that matters here.<sup>1</sup> Further, numerous HTTP protocols have been used over time, and not even plaintiffs claim that the patent requires that the browser work with a particular protocol. Finally, plaintiffs’ objection to Microsoft placing the hypermedia document and the object on a server is inexplicable: plaintiffs insisted that Viola be shown to work in a distributed hypermedia environment, and in such an environment (including the Internet) the document and object will typically be on servers. Plaintiffs appear to be objecting to Microsoft demonstrating that Viola anticipates.

b. *The browser uses “type information” to “identify and locate” an executable application.* Plaintiffs also argue that Viola does not use “type information” to “identify and locate” the executable application (Pl. Br. 25-26, 28),

---

<sup>1</sup> Plaintiffs themselves argued that *capability* is the issue with respect to infringement: “All we have to show is that the product has the accused functionality.” (A101424.) The same must apply to anticipation.

asserting that the object displayed by Viola has associated with it a file name and file path for the executable application and thus that the object somehow does the identifying and locating. But plaintiffs do not dispute that a file name and file path can be “type information” within the meaning of the claims. (Pl. Br. 25-26; *see* A80-81; A101309.) And, contrary to plaintiffs’ suggestion, the “Viola object” does not “fully identif[y] and locate[] the executable application.” (Pl. Br. 26.) Rather, the Viola object contains the file name and file path but *does* nothing with that information; that information is, in the words of the claims, “associated with” the object. Plaintiffs argued, and the district court held in its claim construction ruling, that the fact that the particular type information associated with an object may leave the browser with “very little ... to do in identifying and locating the application” does not mean the browser is not operating as required by the claims. (A80.) Plaintiffs’ expert’s contrary testimony (A100844-47) is irrelevant. That testimony amounts to an effort to create an issue of fact out of what is an issue of law. And creating an issue of fact could not in any event support a ruling that Viola does not anticipate as a matter of law.

c. *The browser “parses” a hypermedia document.* Plaintiffs lastly claim that the DX34 version of Viola does not “parse” the hypermedia document, as required by the claims, because it “uses the separate SGMLS program to do so.” (Pl. Br. 27-28.) This is yet another distraction. Plaintiffs’ cited testimony

establishes *not* that Viola could not parse hypermedia documents, but only that Wei decided (at first) not to write his own parser but to borrow one that was freely available. (A101181-83.) Wei considered the parser, whether his own or the borrowed SGMLS parser, a part of his browser. (A101181.)<sup>2</sup> The law does not require every part of an anticipating device to have been created from scratch by its maker. Wei was as free as Doyle (who used Mosaic (A100720; A100722-23)) to use publicly available code when building his browser.

In the end, plaintiffs have almost no evidence to rebut Microsoft's showing that the DX34 version of Viola anticipates. Plaintiffs' arguments do not come close to establishing that no reasonable jury could find anticipation. Plaintiffs' primary effort to save the district court's ruling from its erroneous reasoning fails.

## **2. DX34 Was Not Abandoned.**

Neither can DX34's exclusion be supported on the ground that Wei "abandoned" his "invention" under section 102(g). The burden was on plaintiffs to present some evidence creating a fact issue regarding abandonment. *Dow Chemical Co. v. Astro-Valcour, Inc.*, 267 F.3d 1334, 1339 (Fed. Cir. 2001).

---

<sup>2</sup> The undisputed record shows that the DX37 version of Viola (A170023) expressly "identifies SGMLS as the parser that's being used" (A101274) and thus teaches the use of that parser.



Plaintiffs presented no such evidence, and Microsoft proved that Wei did not abandon his invention.

Plaintiffs claim that when moving from the DX34 to DX37 versions of Viola, “Wei changed the functionality of DX34” through which the DX34 version was able to work in a hypermedia environment. (Pl. Br. 33-34.) As the testimony cited by plaintiffs demonstrates, however, those “changes” were merely “bugs” that Wei fixed. (A101161-62.) Contrary to plaintiffs’ suggestion (Pl. Br. 33), that testimony is amply corroborated. Microsoft proffered documents showing that Wei continued diligently to work on his browser and to eliminate any “bugs,” which obviously would include an inability to accept certain files it requested from the Internet. (*E.g.*, A2414 (inviting “[b]ug reports” from users).) Indeed, these documents expressly confirm Viola’s ability to work in a distributed hypermedia environment. (A2417 (stating “[t]he ViolaWWW browser could fetch this [ ]application (embedded in HTML or not) from a WWW server, and renders it on the fly.”).) This was a *web browser*; the notion that Wei would distribute alpha and beta releases if it was unable to perform the most elementary function of a browser is absurd. (A2410-19.)

Finally, plaintiffs’ arguments devolve into wordplay when they seize on the fact that Wei changed the name of a “tag” in the Viola code from VOBJF to

<LINK>. Plaintiffs assert that with LINK, the entire “architecture” of the DX34 version of Viola (which employed the VOBJF label) was abandoned. (Pl. Br. 34.) This borders on dishonesty. Wei testified unmistakably that he “took VOBJF ... the exact same functionality, and ... named it LINK.” (A101168.) Plaintiffs’ counsel questioned Wei about functional differences between VOBJF and LINK and learned only that LINK had *additional* capabilities. (A101175.) There is no suggestion *anywhere* in the record that any “differences” between VOBJF and LINK have any relevance to the ’906 claims. (A101173-74.) Moreover, Wei testified without contradiction that, even after he introduced LINK, Viola retained the code supporting VOBJF. (A101168-69.) In short, plaintiffs are asserting that Wei abandoned Viola features that Viola undeniably retained.

Plaintiffs complain that Wei’s testimony discussing the insignificance of the change from VOBJF to LINK is uncorroborated. (Pl. Br. 34.) Plaintiffs are truly grasping at straws, for it is Wei’s testimony on which they rely to argue that the change occurred in the first place. (*Id.*, citing A101172-73; A101175; A101184.) Moreover, the concern addressed by the corroboration requirement — “that a party claiming inventorship might be tempted to describe his actions in an unjustifiably self-serving manner in order to obtain a patent or to maintain an existing patent,” *Singh v. Brake*, 222 F.3d 1362, 1367 (Fed. Cir. 2000) — is not present here because Wei was not attempting to obtain or preserve any patent

rights. Further, Microsoft offered compelling corroborating evidence: *e.g.*, Wei's rapid-response, contemporaneous emails after Doyle's public announcements regarding Doyle's purported invention. *Sandt Technology, Ltd. v. Resco Metal & Plastics Corp.*, 264 F.3d 1344, 1350-51 (Fed. Cir. 2001) (the "most reliable proof" of corroboration is "[d]ocumentary or physical evidence that is made contemporaneously with the inventive process" because "the risk of litigation-inspired fabrication or exaggeration is eliminated."). Those emails, consistent with Wei's testimony, reflect Wei's view that Viola had the '906 functionality since before May 1993 (when it employed the VOBJF tag) and continued to have it even after Doyle's announcement (when Viola supported both tags). (A2435; A2440; A2452; A2455.) Plaintiffs cannot claim that Wei generated those emails for any "litigation-inspired" reason. Finally, Microsoft was prepared to offer expert testimony that would have further corroborated Wei's testimony regarding the immaterial change from VOBJF to LINK. (A2343-44.) Plaintiffs are simply ignoring inconvenient evidence.

In the end, DX34 should have been admitted into evidence to allow Microsoft to establish prior invention under section 102(g). Plaintiffs sufficiently disoriented the district court that the court excluded DX34 on a ground that not even plaintiffs try to defend. A new trial is necessary.

**3. DX34 Was Independently Admissible To Establish Microsoft's Section 102(b) Prior Public Use Defense.**

According to plaintiffs, but not the district court, DX34 was inadmissible to show public use because the demonstration to Sun Microsystems was not performed in a distributed hypermedia environment. (Pl. Br. 29.) That addresses the wrong question. What matters is whether DX34's ability to work in a distributed hypermedia environment was *disclosed* to Sun. Plaintiffs infer that it was not merely because the demonstration was local. That inference is unsupportable.

As an initial matter, the Sun engineers were interested in Viola because it was a *browser*, which by definition is intended for use in a distributed hypermedia environment. (A2337; A2373 (Sun wanted a "system that can go anywhere and from anywhere be on the net").) And Microsoft was prepared to offer testimony that Wei had explained to the Sun engineers, before performing the demonstration, that the Viola browser he was demonstrating worked in a distributed hypermedia environment. (A2337.)

Even if plaintiffs were correct that there was no reason to believe that DX34's ability to work in a distributed hypermedia environment had been expressly disclosed to the Sun engineers, plaintiffs' argument would still fail. Plaintiffs rely on the same proposition of law rejected in *Netscape*

*Communications Corp. v. Konrad*, 295 F.3d 1315 (Fed. Cir. 2002). In *Netscape*, the patentee, like plaintiffs here, argued that there was no “public use” because the demonstration to two individuals knowledgeable in the computing field failed to “disclose every limitation of [the claimed] invention.” *Id.* at 1321. This Court rejected the patentee’s effort to require demonstration of all elements of the claim for the public use bar to apply. “Section 102(b) may bar patentability by anticipation if the device used in public includes every limitation of the later claimed invention, or by obviousness if the differences between the claimed invention and the device used would have been obvious to one of ordinary skill in the art.” *Id.* Because the single element of the claimed invention that apparently had not been demonstrated in *Netscape* would have been obvious to the skilled computing personnel to whom the invention was demonstrated, the public use bar applied. *Id.* Likewise, here, because the single claimed element that plaintiffs assert was not demonstrated — the browser’s ability to work in a distributed hypermedia environment — would have been obvious to engineers from Sun Microsystems, the public use bar should apply.

Plaintiffs suggest *Netscape* should not apply because Wei is not asserting patent rights to his invention. It is true that a *secret* use by a third party may not trigger the public use bar. *W.L. Gore & Associates, Inc. v. Garlock, Inc.*, 721 F.2d 1540, 1550 (Fed. Cir. 1983). But it is *Gore*, not *Netscape*, that is

inapplicable here. In *Gore*, a manufacturer was selling a product made by a patented process while under a contractual obligation to maintain the secrecy of the machine employing the process. Only the product, not the process claimed in the patent, was publicly disclosed. *Id.* This case involves a public demonstration of the invention itself — not merely its product — by the individual who actually invented it first. Indeed, it would be bizarre to leave Wei's invention available for patenting by another more than one year *after* he publicly demonstrated it to one of the leading companies in the industry merely because Wei wanted to commit his invention to the public. Plaintiffs never explain how such a rule would “serve[] the policies of the patent system,” *Netscape*, 295 F.3d at 1320.

**B. The Jury Should Have Been Permitted To Consider The Viola Evidence That Was Admitted.**

Plaintiffs' defenses of the JMOL ruling substantially mimic their arguments that the DX34 version of Viola does not, as a matter of law, anticipate. To the extent those arguments have been addressed above, the responses will not be repeated.

Plaintiffs raise two issues unique to the DX37 version of Viola. First, plaintiffs claim that Microsoft admitted that DX37 does not, without “gap-filling,” anticipate the invention. (Pl. Br. 17-18, 35-36.) There was no “admission.”

Plaintiffs seize upon a misstatement by one of the lawyers for Microsoft, in the

immediate aftermath of the court's surprising exclusion of DX34, and ignore the fact that Microsoft consistently made clear, before and after the cited exchange (e.g., A101127-32; A101253; A101189; A101354), that DX37 was offered for both anticipation and obviousness. The court and plaintiffs' counsel recognized that Microsoft had not "admitted" non-anticipation. (E.g., A101253; A101347; A101362-63.) Plaintiffs' assertion is gamesmanship.

Second, plaintiffs emphasize one "bug" particular to DX37: the difficulty this version of Viola had operating in a distributed hypermedia environment when the object to be displayed was not local. (Pl. Br. 21-22.)

Microsoft explained that this "bug" was irrelevant to anticipation because the '906 patent requires that *only* the hypermedia document, *not* the object, be retrieved from a remote location. (Microsoft Br. 45.) The claims are *not* limited to displaying remote objects, which plaintiffs do not dispute. Instead, plaintiffs take issue with Microsoft's expert's testimony (A101284) that the DX37 version of Viola would work if the object were local. (Pl. Br. 23.) Plaintiffs assert that locating the object locally "would require *new changes* to the VOBJF architecture." (Pl. Br. 23.) Plaintiffs had the opportunity to cross-examine Microsoft's expert on this issue, but failed to elicit *any* testimony supporting their assertion. It has no basis in the record.

Plaintiffs disagree with the testimony offered by Wei and corroborated by Dr. Kelly's expert analysis. (Pl. Br. 23-25.) Disparaging Dr. Kelly's testimony as "conclusory" does not make it so. Plaintiffs would be free to argue to the jury that Dr. Kelly's testimony was unpersuasive. But the jury should have been permitted to consider his detailed testimony regarding where in the Viola code each limitation of the claims was taught. (A101274-79; Microsoft Br. 43-45.)

Finally, plaintiffs argue that obviousness was properly taken from the jury because Microsoft presented an anticipation defense (which plaintiffs earlier say Microsoft abandoned). (Pl. Br. 17-18, 35-36.) But it is common for defendants to assert *both* anticipation and obviousness. Here, Microsoft argued that DX37 taught all the limitations of the claims, and when plaintiffs asserted to the contrary, Microsoft cited evidence establishing that what plaintiffs claimed were missing limitations — including a browser's ability to work in a distributed hypermedia network environment — would be obvious to those skilled in the art. (A100721; A101283-84.)

## **II. A NEW TRIAL ON INEQUITABLE CONDUCT IS REQUIRED.**

Plaintiffs do not dispute that Doyle knew what Wei claimed to have invented, began investigating Wei's claims, knew where to get a version of the Viola code, but chose not to do so. (A29-33; A101717-24.) Microsoft pointed out



this Court's admonition that a patent applicant may not "cultivate ignorance," as Doyle did. *FMC Corp. v. Hennessy Indus.*, 836 F.2d 521, 526 & n.6 (Fed. Cir. 1987). Strikingly, without discussing *FMC Corp.*, plaintiffs defend Doyle's decision not to inform the PTO about Viola based on his cultivated ignorance. (Pl. Br. 39 (emphasizing that Doyle never had the Viola code).)

Plaintiffs also emphasize the district court's finding that Doyle did not intend to deceive the PTO when he concealed what he knew about Viola. (Pl. Br. 40.) Again, plaintiffs simply ignore Microsoft's argument: this very finding was driven by the court's manifest misunderstanding of the prior art status of Viola, in particular its view that each version was separate and distinct. The court failed to recognize the materiality of what Doyle withheld, and thus could not properly assess his intent. As this Court has held, when "withheld information is material and the patentee knew or should have known of that materiality," an inference of intent to mislead arises. *Bristol-Myers Squibb Co. v. Rhone-Poulenc Rorer, Inc.*, 326 F.3d 1226, 1239 (Fed. Cir. 2003). Plaintiffs ignore this principle.

Had the district court appreciated the significance of Viola, it could not have found that Doyle's intentions were pure when he withheld the Viola information. A new trial on inequitable conduct is warranted.

### III. THE DISTRICT COURT'S CLAIM CONSTRUCTION WAS WRONG AS A MATTER OF LAW.

#### A. The Court Construed "Executable Application" Contrary To Its Plain Meaning And The Intrinsic Evidence.

The district court's construction of "executable application" to include components such as DLLs was contrary to the term's ordinary meaning, as well as the specification and prosecution history. (Microsoft Br. 51-56.) Although plaintiffs consistently mischaracterize the intrinsic evidence, they ultimately point to nothing that rebuts the heavy presumption that the ordinary meaning should apply. *See Texas Digital Sys., Inc. v. Telegenix, Inc.*, 308 F.3d 1193, 1204 (Fed. Cir. 2002).

1. *Ordinary meaning*: The ordinary definition of "application" (by reference to "computer program") "usually implies a self-contained entity, as opposed to a routine or a library." *Microsoft Press Computer Dictionary*, 2d ed., 1994, pp. 23-24, 90. Furthermore, "executable," when used with "application" or "program," requires that the application be "ready to run." *See id.* at 153 ("A computer program that is ready to run.... *The user* does not have to alter the program in any way *before being able to run it.*") (emphasis added). An application is not ready to run if the user must first launch another application. Components such as DLLs — that can run only if another application is first launched and running — are not "executable applications."

Plaintiffs nevertheless assert that the district court correctly construed “executable application” to encompass “*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 the data.” (Pl. Br. 42 (quoting A89) (emphasis added).) Plaintiffs’ position has several major flaws.

First, plaintiffs’ construction fails to give effect to *both* words in the phrase “executable application,” contrary to the requirement that every word in a claim be given effect. *See, e.g., Wright Medical Tech., Inc. v. Osteonics Corp.*, 122 F.3d 1440, 1444 (Fed. Cir. 1997); *see also Texas Instruments Inc. v. United States Int’l Trade Comm’n*, 988 F.2d 1165, 1171 (Fed. Cir. 1993). Plaintiffs correctly assert that modifiers cannot be *added* to claim terms (Pl. Br. 42), but neither can they be deleted. Plaintiffs attempt to delete “executable” — presumably because they believe it is not “germane” to the invention (*see* Pl. Br. 45) — but post-issuance editing is not permitted.<sup>3</sup>

---

<sup>3</sup> Even if it were appropriate (which it is not) to discard claim language based on what is “germane” to an invention, the record confirms that a standalone application is germane. The source code attached to the ’906 patent can invoke only standalone applications, not components such as DLLs (A100124; A100209-10), and the inventors, in distinguishing the OLE prior art, told the PTO that it would have been extremely difficult to modify Mosaic to work with components (A150344-48).

Second, Microsoft's expert never agreed that "executable applications" include DLLs, as plaintiffs falsely assert. (Pl. Br. 43.) Microsoft's expert was discussing "executable application" as it had been erroneously defined by the district court, not his own construction. (See A101306.)

Finally, plaintiffs point to the fact that the dictionary definition of "executable program" states that the term "*usually* implies a self-contained entity, as opposed to a routine or library." (Pl. Br. 43 (emphasis added).) According to plaintiffs, this "usual" definition should not apply because "[o]ther versions of Microsoft's dictionary offer broader understandings." (Pl. Br. 44.) Yet plaintiffs fail to explain how the particular definition they cite applies to the claim language and intrinsic record here.<sup>4</sup> Moreover, the "other" Microsoft dictionary was published three years after the '906 application was filed. Because claims are construed "as they were understood at the time the patent application was filed," *Plant Genetic Sys., N.V. v. DeKalb Genetics Corp.*, 315 F.3d 1335, 1345 (Fed. Cir. 2003), such after-the-fact definitions are irrelevant.<sup>5</sup>

---

<sup>4</sup> See *Phillips v. AWH Corp.*, 2004 WL 1627271 (Fed. Cir. July 21, 2004) (ordering *en banc* rehearing of claim construction issues).

<sup>5</sup> Plaintiffs also cite the 1994 *IBM Dictionary of Computing* (Pl. Br. 45), but ignore the most relevant definition, which states that, in IBM's version of UNIX, an "executable program" is "a program that can be run as a self-contained procedure." *IBM Dictionary of Computing*, pp. 18, 250.

2. *The specification:* The specification confirms that the inventors used “executable application” in its ordinary sense of standalone applications. *Every* disclosed embodiment uses a self-contained application. (Microsoft Br. 53.) Moreover, the specification consistently describes communications between the browser and executable application as “interprocess communications” (*see, e.g.*, A150015, col. 7, lns. 1-4; A150016, col. 9, lns. 7-10), which are communications between two processes each running in its *own* memory space — *i.e.*, standalone applications (*see* A100203-07).

Plaintiffs do not dispute that *all* of the embodiments in the specification use standalone applications. Instead, they cite the inapposite axiom that “claims are not limited to preferred embodiments.” (Pl. Br. 46.) Where, as here, *all* of the disclosed embodiments are consistent *only* with the ordinary meaning of the claim language, the disclosure confirms that the ordinary meaning should apply. *See, e.g., Toro Co. v. White Consolidated Indus., Inc.*, 199 F.3d 1295, 1301 (Fed. Cir. 1999); *Laitram Corp. v. Morehouse Indus., Inc.*, 143 F.3d 1456, 1463 (Fed. Cir. 1998).

Plaintiffs also point to passages that indicate that the “specification and drawings are ... to be regarded in an illustrative rather than a restrictive sense.” (Pl. Br. 46 (citing A150019 & A150015-17).) This boilerplate has nothing to do

with the meaning of “executable application” and cannot overcome the heavy presumption in favor of ordinary meaning. *Cf. Fromson v. Anitec Printing Plates, Inc.*, 132 F.3d 1437, 1447 (Fed. Cir. 1997) (where only non-dissolving electrolytes were shown, “boilerplate statement” that other “suitable liquids” can be used did not encompass specific dissolving electrolytes).

Plaintiffs similarly cite the specification’s statement that “[a]ny manner of applications program may be specified by the TYPE element so that other types of applications, such as a spreadsheet program, database program, word processor, etc. may be used with the present invention.” (Pl. Br. 45 (quoting A150018).) How plaintiffs believe this passage, which refers *only* to standalone applications, supports their construction is a mystery.

Plaintiffs, seeking *any* reference to non-standalone applications, also point to the specification’s statement that the *browser application* may be “implemented by one or more software routines, processes, subroutines, modules, etc.” (A150018, col. 13, lns. 60-63.) Plaintiffs assert that this statement, in a context having nothing to do with the “executable application” limitation, should apply “equally” to “executable application.” (Pl. Br. 46.) The assertion is worse than baseless. At most, this passage demonstrates that the patentees knew how to

refer to non-standalone applications, and chose *not* to do so when referring to the “executable application.”<sup>6</sup>

3. *The prosecution history*: Finally, plaintiffs distort the PTO record, which, read in full, supports a construction that requires standalone applications.

Plaintiffs begin with a blatant mischaracterization. They claim the “examiner defined executable application” to include “‘launchable’ program *instructions/codes*.” (Pl. Br. 47 (quoting A150083).) This was no definition of the claim term “executable application” because that claim term did not exist at the time of this office action. (See A150054 (claiming “controllable application”).)

Plaintiffs then argue that the applicants distinguished the prior art based not on the prior art’s use of components but rather because it did not enable interactivity. (Pl. Br. 49-50.) But plaintiffs ignore the applicants’ initial *failed* effort to distinguish the prior art based on interactivity. (A150521.) It was *after* they failed to win their patent on that basis that they distinguished the prior art based on the prior art’s use of non-standalone applications. (A150344; Microsoft Br. 54-55.) And the Examiner specifically relied on this distinction — the claimed invention’s use of standalone applications — to allow the patent. (A151031.)

---

<sup>6</sup> Plaintiffs’ reliance on testimony that one can “build an application with DLLs” (see Pl. Br. 43) is similarly misplaced. The fact that an application can be built

Plaintiffs try to save their construction from the Examiner's own words by arguing that the Examiner was limiting the prior art only to the "*particular kind*" of non-standalone applications taught in Koppolu. (Pl. Br. 50.) But the only category of non-standalone applications they suggest the Examiner was excluding are those that "did not allow interactive processing of the object." (Pl. Br. 50.) In short, this argument is yet another attempt to assert the distinction based on interactivity that the Examiner rejected.

In the end, the Examiner specifically referred to the cited art's use of non-standalone applications in his statement allowing the claims. Plaintiffs ask this Court to transform that reference into a reference to interactivity, even though the Examiner had rejected that basis for distinguishing the claims and never suggested that he had changed his mind. A more self-serving misreading of the prosecution history is hard to imagine.<sup>7</sup>

\* \* \*

---

*using DLLs* says nothing about what the application is *after* it is built.

<sup>7</sup> Equally self-serving is plaintiffs' reference to Doyle's assertion during prosecution that ActiveX controls incorporate features of his invention. (Pl. Br. 47-48.) There is no suggestion that the Examiner was told or understood that ActiveX controls are components as opposed to standalone applications. (See A100224-25.) In any event, Doyle's after-the-fact statements cannot expand the scope of claim language. See *Multiform Desiccants, Inc. v. Medzam, Ltd.*, 133 F.3d 1473, 1478 (Fed. Cir. 1998).



Plaintiffs do not dispute that ActiveX controls, applets, and plug-ins are all components and that they are not covered by the '906 claims under what Microsoft contends is the correct construction. That construction should be adopted, and Microsoft should be granted judgment as a matter of law, *see Electro Scientific Indus., Inc. v. Dynamic Details, Inc.*, 307 F.3d 1343, 1350 (Fed. Cir. 2002), or at a minimum a new trial, *see Ecolab, Inc. v. Paraclipse, Inc.*, 285 F.3d 1362, 1373 (Fed. Cir. 2002).

**B. The District Court Erred In Construing The “By the Browser” Limitation.**

The district court also erred in instructing the jury concerning the limitation requiring that the identifying and locating functions be performed by the browser. (*See* Microsoft Br. 56-60.) At the *Markman* stage, the court correctly held that those functions “must be performed by the browser, not the operating system as in Koppolu’s OLE.” (A88.) However, when instructing the jury, the court inexplicably refused to explain that OLE’s use of the operating system marks the outer bounds of the claims. This error allowed the jury to read the claims to encompass the very technology that the applicants had distinguished, and that Microsoft had demonstrated that its products used.

Plaintiffs’ carefully-crafted response ignores Microsoft’s argument.

Plaintiffs emphasize that the inventors “did not disclaim any operating system role

in the prosecution history” and deny that the inventors “disclaimed any use of the operating system by the browser.” (Pl. Br. 52-53.) The issue was never *whether* the operating system plays a role, but *what* is the limit on its role.

As the district court originally, and correctly, concluded, OLE provides the limit. The applicants “argu[ed] that both Khoyi and Koppolu-OLE were different [from their invention] because they were operating system-dependent.” (A87-88; *see also* A150902; A150340; A150342-44.) Plaintiffs’ selective quotation from the prosecution history cannot obscure that fact. (Pl. Br. 53-54.)

Recognizing OLE as the outer limit of the claims does not, as plaintiffs assert, read a preferred embodiment out of the claims. (Pl. Br. 51.) The embodiment plaintiffs cite itself makes clear that *the browser* identifies and locates the executable application (*see* Pl. Br. 51 (citing A150019)) and thus is fully consistent with Microsoft’s construction, because the browser can consult outside resources as long as it — the browser — does the identifying and locating. In OLE, in contrast, the operating system itself determines what application to launch. (*See* Microsoft Br. 13-15, 56-58.) No embodiment in the patent operates as OLE does.

All of this was lost in the jury instruction. Plaintiffs do not suggest that the instruction adequately communicates that OLE provides the limit of the claims. (Pl. Br. 54-55.) Microsoft's proposed instruction would have made that clear: "[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.) Plaintiffs never deny that the instruction as given permitted the jury to find infringement even though the accused products function "as in Koppolu's OLE." (A88.) Whatever else the instruction said, if it did not preclude *that* possibility, the instruction was erroneous.

Plaintiffs absurdly contend that Microsoft cannot establish prejudice. (Pl. Br. 54.) Relying on the court's pretrial construction, Microsoft emphasized that its products performed just as OLE. The court abandoned that construction, refusing to instruct the jury that the identifying and locating cannot be done as in OLE. Plaintiffs' counsel magnified the prejudice by encouraging the jury to find infringement even though Microsoft's accused products function just as does Koppolu's OLE, by using the operating system to identify and locate. (*See* A101457; A101433; A101297; *see also* Microsoft Br. 60.) In short, Microsoft's defense was turned into plaintiffs' offense.

Finally, plaintiffs assert that any error is harmless because Microsoft admitted that IE's CODEBASE functionality identifies and locates some executable application. (Pl. Br. 56-57.) Microsoft's witness, Mr. Wallent, testified that IE can obtain a file name and file path for an application to determine whether a file downloaded from a remote location has been locally installed. (See A101124; A101301.) There is no evidence that IE ever does *anything* with the file name and file path relevant to the '906 functionality. It does not use them to execute the application; it does not pass them on to the operating system. In fact, after the remote application is downloaded and installed on the local computer, the operating system still must identify and locate the application to launch the newly-installed control in the same manner as preinstalled controls. (A101301-02; A101288-89.) In short, the "identify and locate" operations of the claims still remain to be performed by the operating system *after* IE retrieves a remote application and it is installed locally. (See A101301.) At a minimum, a properly-instructed jury should be permitted to consider the CODEBASE evidence.

This Court, accordingly, should enter judgment for Microsoft, *see Electro Scientific*, 307 F.3d at 1350, or order a new trial, *see Ecolab*, 285 F.3d at 1373.

**IV. THE DISTRICT COURT ERRED AS A MATTER OF LAW BY HOLDING THAT FOREIGN SALES INFRINGE UNDER SECTION 271(f).**

Relying on the language and history of 35 U.S.C. § 271(f), Microsoft demonstrated that the district court erred in interpreting section 271(f) to include as an act of infringement the use of *information*, not parts, sent from the U.S.

(Microsoft Br. 60-65.) Microsoft's interpretation of the statute is confirmed in the recent decision in *Pellegrini*, 375 F.3d 1113. Plaintiffs do not even attempt to distinguish *Pellegrini*. (See Pl. Br. 58.) Nor could they. *Pellegrini* makes clear that section 271(f) "refers to *physical supply* of components, not simply to the supply of instructions." *Pellegrini*, 375 F.3d at 1118 (emphasis added).

Microsoft does not supply any physical components of foreign-made products. Rather, the "golden master" that Microsoft sends from the U.S. is a medium for conveying information to foreign manufacturers, the same as a blueprint, mold, or template. When that information, the code reflected in the pits and lands on the "golden master," is given to OEMs for replication abroad, the "golden master" — the only item physically present in and sent from the U.S. — remains unchanged. The fact that the code on the "golden master" represents an important contribution to the foreign products is irrelevant. No physical part of the

“golden master” becomes part of the foreign-made products.<sup>8</sup> Section 271(f) therefore cannot apply.

Plaintiffs accuse Microsoft of seeking special treatment for software. (See Pl. Br. 58-59.) But it is plaintiffs that seek a software-only rule. As they must, because their position — that section 271(f) reaches the *information* conveyed via the “golden master” — would work a revolution if generally adopted. Under plaintiffs’ interpretation, any patented product made abroad using U.S.-made prototypes or blueprints or molds would fall within section 271(f)’s expanded reach. There is no indication that Congress intended to create such sweeping extraterritorial reach for the patent laws when it closed the loophole created by *Deepsouth Packing Co. v. Laitram Corp.*, 406 U.S. 518 (1972), which permitted *physical* parts of a patented invention to be manufactured in the United States but assembled abroad without infringing. See *Pellegrini*, 375 F.3d at 1116. Nor is there any basis in either the statutory text or legislative history for limiting the expansive effect of plaintiffs’ proposed interpretation by creating a software-only rule. This Court, accordingly, should reject that position. See *Bayer AG v.*

---

<sup>8</sup> Plaintiffs’ assertion that the code is “installed” on, rather than copied to, computer usable media abroad (see Pl. Br. 61) is meaningless wordplay. Whether the process is called “installation” or “replication,” the fact remains that nothing is physically removed from the “golden master” and placed onto hard drives abroad.

*Housey Pharms., Inc.*, 340 F.3d 1367, 1376-77 (Fed. Cir. 2003); *Pellegrini*, 375 F.3d at 1116-18.

In sum, when section 271(f) is properly construed, products made and sold outside the U.S. cannot infringe, which requires a 64 percent reduction in the award. (See A2320.) Plaintiffs' claim that the award should be reduced by 42.4 percent is misguided. (Pl. Br. 62.) Their calculation rests on data that categorize sales based on the *billing address* of the OEM. But all that matters, and what their data fail to address, is where the units are *made and sold*. (See A151550-79; A151663-64; *see also* A2319.) As Microsoft's offer of proof shows,<sup>9</sup> 226,639,360 units were made abroad and 127,484,640 in the U.S. (*see* A2320), requiring that the award 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.

---

<sup>9</sup> Plaintiffs' complaint that Microsoft presented no evidence at trial concerning foreign sales is baseless. (Pl. Br. 62.) Whether Microsoft might have had a reason to submit geographic sales data for some other purpose, but chose not to do so, is beside the point. There is no dispute that Microsoft would have submitted such data had the court properly interpreted section 271(f). The court did not. Therefore, Microsoft properly submitted an offer of proof. (A2318-23.) Plaintiffs do not dispute the accuracy of Microsoft's data.

The judgment holding the patent not invalid or unenforceable should be vacated and remanded for new trial. 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:

T. Andrew Culbert  
Microsoft Corporation  
One Microsoft Way  
Building 8, N/203  
Redmond, WA 98052  
425-882-8080

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

Attorneys for Microsoft Corporation