No. 05-1056

IN THE
Supreme Court of the United States

MICROSOFT CORPORATION,

Petitioner,

v.

AT&T CORP.,

Respondent.

ON WRIT OF CERTIORARI TO THE UNITED STATES COURT OF APPEALS FOR THE FEDERAL CIRCUIT

BRIEF OF U.S. PHILIPS CORPORATION AND PHILIPS ELECTRONICS NORTH AMERICA CORPORATION AS *AMICI CURIAE* IN SUPPORT OF RESPONDENT

JACK E. HAKEN
EDWARD BLOCKER
PHILIPS INTELLECTUAL PROPERTY & STANDARDS
345 Scarborough Road
Briarcliff Manor, NY 10510
(914) 945-6000

JOHN M. DIMATTEO
Counsel of Record
EUGENE CHANG
WILLKIE FARR & GALLAGHER LLP
787 Seventh Avenue
New York, NY 10019
(212) 728-8000

Counsel for Amici Curiae U.S. Philips Corporation and Philips Electronics North America Corporation

January 23, 2007

QUESTIONS PRESENTED

(1) Whether digital software code—an intangible sequence of "1's" and "0's"—may be considered a "component[ ] of a patented invention" within the meaning of Section 271(f)(1); and, if so,

(2) Whether copies of such a "component[ ]" made in a foreign country are "supplie[d] . . . from the United States."

TABLE OF CONTENTS

TABLE OF AUTHORITIES

CASES

STATUTES

MISCELLANEOUS

INTEREST OF AMICI CURIAE[1]

With this appeal, software companies (and similarly situated "new economy" companies) are seeking to have this Court create an exception for software in 35 U.S.C. § 271(f). Such an exception would unfairly favor the software industry over traditional electronics hardware companies, such as Philips. Furthermore, the interpretation sought by Microsoft and its amici ignores the proper measure of damages, which would include all foreign sales resulting from the infringing act.

Amici curiae U.S. Philips Corporation and Philips Electronics North America Corporation are part of the Philips worldwide family of companies ("Philips"). Like other parts of the Philips family, the amici curiae are wholly owned by the Dutch parent company, Koninklijke Philips Electronics N.V.

Philips has been inventing and manufacturing electronic and electrical products for over 115 years, and is one of the largest users of the patent systems in the United States and other industrialized countries. Last year Philips filed U.S. patent applications for about 3,000 new inventions. Scientists and engineers at our U.S. laboratories have made pioneering advances in the fields of high definition television, optical CD and DVD recording, digital cellular telephones, medical imaging, and digital rights management.

---

[1] Amici curiae submit this brief with the written consent of the parties, as reflected in letters filed with the Clerk of this Court. No counsel for a party authored this brief in whole or in part, and no person or entity other than amici curiae and their counsel has made a monetary contribution to its preparation or submission.

Section 271(f) allows us to protect our intellectual property rights when American-based companies export specialized components and parts kits for assembly in countries that currently do not have useful patent enforcement systems, particularly China, Taiwan, Korea, and India.

Section 271(f) was enacted by Congress more than 20 years ago to plug a loophole that had allowed American companies to avoid liability for patent infringement by creating the essential components of a patented invention in the U.S., but assembling the device overseas. In the decision currently on appeal, the United States Court of Appeals for the Federal Circuit confirmed that Section 271(f) applies to software components as well as hardware components. If this Court reverses that decision and exempts software components from the protection of Section 271(f), it will weaken the patent protection for many of our most innovative products.

For example, Philips invented the optical recording technology that enabled development of the CD and DVD industries. When Philips started licensing this patented technology, the technology was implemented as hardware circuits in disc drives and recorder sets. Today the same technology is implemented through PC software that is distributed over the Internet and through firmware[2] that is stored in semiconductor memory chips.

Another example is speech and image coding technology. Philips invented important speech and image

[2] Firmware is software that is imbedded into an electronic device. It can be installed and upgraded like any other software.

coding methods and devices that we license to manufacturers of cellular telephones, digital cameras, and set-top boxes.[3] Today, American-based companies are the primary developers of the application and operating system software that controls the hardware and implements the coding methods. These companies often export software code on master disks or as firmware on semiconductor memory chips, and they distribute upgrade software over the Internet and cable television networks.

The list of pioneering hardware inventions now implemented through software goes on and on. If this Court reverses the decision currently on appeal, it will create an exception for software code that will allow infringers to avoid paying appropriate damages by implementing their device in software and moving the final assembly step—installation of the software—overseas. That result is neither fair nor justifiable.

SUMMARY OF ARGUMENT

On its face, this would appear to be an appeal centered on the abstract statutory construction of two words: "component" and "supply." Appearances are deceiving. In reality, the interpretation argued by Microsoft and its amici would create an exception that favors the software industry over other industries, including traditional electronics hardware companies.

This case involves economic and technical issues at the heart of the new economy, particularly the convergence

[3] Philips holds patents on codecs that are similar to the patents involved in the case currently on appeal.

of hardware and software devices. Over the past decade, software has become increasingly interchangeable with hardware, particularly in the field of consumer electronics.

Traditional electronics hardware companies have relied on Section 271(f) to protect their intellectual property for over 20 years. Software companies typically have also been proponents of strong patent protection—at least for software and business methods. Now that they compete directly with electronics hardware companies, however, these same software companies seek to avoid paying proper damages by creating an exception for software code from the protection of Section 271(f). If this Court rules in their favor, it will favor software companies over hardware companies in this emerging competitive landscape.

Microsoft and its amici raise two main arguments to try to convince this Court to exclude software code from the scope of Section 271(f). First, they argue that executable software code can never be a "component" of a patented invention, even though the software is an essential part of most electronic devices. Second, they argue that the software is not "supplied" from the United States, even though the executable code is created here in its final form, and then sent overseas with the intent that it be installed without modification onto the infringing products. Neither of these arguments has merit.

To argue that software is not a component under Section 271(f), Microsoft and its amici present this Court with a series of far-fetched hypothetical situations and bizarre "molecule" tests. These arguments blithely ignore how software is developed, used, and distributed in the real world.

At minimum, when software substitutes directly for, and has the same technical effect as, a piece of specialized hardware, it must be a component just as is the specialized hardware.

Like other software companies, Microsoft intended to supply the executable software code at issue in a standard, customary manner. Everyone with a "Windows" certificate of authenticity knows that the executable software code on that computer was supplied by Microsoft, regardless of how many secondary copies were made during installation and assembly.

Lost in the pages of arguments made by Microsoft and its amici is the real issue in this case: the proper measure of damages. Once a company supplies a software component to foreign installers in the same way that it supplies that component to U.S. installers, the proper measure of damages includes all the consequent foreign sales. It is irrelevant whether the company supplies the component on a golden master disk or over the Internet. When a software company supplies the executable code that is used to create multiple infringing devices, it should be held liable for all attendant damages flowing from that act. Thus, it should pay damages for each of the infringing devices sold abroad.

Contrary to the arguments of Microsoft and its amici, software companies do not face endless liability for infringing devices that are created abroad. Their liability is bounded by their own intent—when they send software abroad with the intent that it be installed on multiple infringing devices, they owe damages for all of those devices, no more, no less. Both patent law and copyright law already allow damages for foreign sales involving

multiterritorial acts of infringement. There is no reason to create an exception for Section 271(f).

Microsoft and its amici also argue that the entire U.S. software industry will move its research and development overseas if they have to pay damages for infringing foreign sales under Section 271(f). This scenario—positing a wholesale exodus based on liability under a single section of the infringement statute—is implausible at best. In fact, new economy software companies benefit from strong intellectual property protection and are not likely to move to countries with weaker intellectual property systems. The patentability of software has been a contentious issue in Europe, and it is hard to believe that U.S. software companies will move to countries where they might not be able patent their software at all, just to sell a product abroad that they cannot sell in the U.S. It is even less likely that they will move to countries with less-developed intellectual property regimes, such as China.

ARGUMENT

I. If This Court Exempts Executable Software Code From Section 271(f), It Would Improperly Favor Software Companies Over Electronics Companies In The Emerging Technology Landscape

The worlds of electronic hardware and software are colliding. Traditional electronics hardware companies, like Philips, have relied on Section 271(f) for over 20 years to provide proper protection of their intellectual property from other hardware companies. Over the past decade, however, competition in the electronics industry has shifted to include "new economy" companies, whose primary outputs are intangible, such as computer code, rather than physical goods. *See* William M. Landes & Richard A. Posner, *The Economic Structure of Intellectual Property Law* 390 (2003). The new economy includes companies like Microsoft, Intel, AOL, and Amazon.com. *Id.*

Now that new economy software companies compete directly with traditional electronics companies, they want this Court to carve out an exception from Section 271(f) for executable software code. If this Court does so, it will improperly favor software companies over hardware companies in the emerging technology landscape.

There are two major changes that have affected the electronics industry during the past decade and that make Section 271(f) so important to hardware companies. The first change is a well-recognized shift of high-volume electronics manufacturing from the United States to overseas contract facilities. Although the United States is no longer a competitive site for manufacturing this equipment, the U.S.

remains a primary locale for the development of technology and essential components that are exported overseas for assembly. Royalties on exports of patented technology are a significant reason why U.S. innovation continues to flourish and why the U.S. domestic research establishment remains competitive in the global marketplace.

The second change is perhaps not as visible as the first. In 1990, almost all consumer electronic devices were implemented in separate specialized sets as combinations of hardware circuit elements. For example, televisions, VCRs and telephone answering machines were produced and marketed in separate boxes and often via different selling channels. Today those same technical functions are usually implemented in software that is installed on a small number of common multiuse hardware platforms. Personal computers and PDAs use software to provide audio and video recorder functionality. Cellular telephones contain software functionality for cameras, music players and, soon, television players.

In this context, the boundary between hardware and software becomes fuzzy. Software and firmware code effectively reconfigures generic hardware circuits to perform technical functions that were formerly achieved in fixed circuitry. Hardware and software now "are practically interchangeable in the field of computer technology. On a functioning computer, software morphs into hardware and vice versa at the touch of a button. In other words, software converts its functioning code into hardware and vice versa." *Eolas Techs. Inc. v. Microsoft Corp.*, 399 F.3d 1325, 1339 (Fed. Cir. 2005) (internal citation omitted).

Failing to apply Section 271(f) to executable software or firmware code will allow infringers of patents on electronic devices to escape liability simply by implementing their design in executable software or firmware code and installing the code overseas.

In the past, if an electronics company manufactured specialized hardware components in the United States and shipped them overseas for combination with other components to create an infringing device, that company was clearly liable under Section 271(f). Today, a software company could take the same functionality and implement it through software or firmware code, put the executable code onto a golden master disk, and ship the code overseas to be installed onto hardware.

The end result is the same—overseas assembly of an infringing product from components made in the United States. Accordingly, the U.S. creator of an essential part of an invention, whether it is implemented in hardware or software, should be liable under Section 271(f) in both circumstances. If this Court decides otherwise, then Section 271(f) effectively will not apply to any industry where the line between hardware and software has blurred. There is no reason to create such an exception for software in Section 271(f).

II. Executable Software Code, A Key Component Of Most Electronic Devices, Is Supplied When It Is Furnished To A User In An Intended And Customary Manner

A. Executable Software Code Is As Much A Component of Electronic Devices As Hardware

Given the convergence of electronic hardware and software in the new economy, it would be ironic for the Court to artificially redraw a line that advances in technology have already erased. Section 271(f) does not limit "components" to machines or physical structures. Executable software or firmware code can be a component of a patented invention for purposes of Section 271(f). In fact, executable software code is such an essential part of modern electronic devices that it is the *primary* component of many products today.

This does not mean that every computer program is a component of a patented invention. But a program that has the same technical effect as an electronic hardware component surely is. In particular, there are two factors that illustrate that *executable* software or firmware code is in fact a component of a patented invention.

First, executable code is distributed in its final form such that it cannot be changed. The software developer designs the software in the form of source code, and then fixes it in an executable form by compiling it. The act of compiling manufactures the executable code. In order to modify the executable code, it must be decompiled, modified, then recompiled—a process similar to using a

sample to manufacture new copies of a gear. Although the software developer may allow the installer to customize certain parameters, the installer is *not* allowed to modify the executable code. For example, Microsoft requires original equipment manufacturers ("OEMs") to attach a Certificate of Authenticity to each fully assembled computer system.[4] This certificate assures customers that they have acquired "**genuine Microsoft Windows software**."[5] Thus, each consumer receives the same executable Windows software code that was compiled in the United States, regardless of where the computer hardware was manufactured or how many secondary copies were created along the way.

Second, executable code is what gives technical hardware its functionality. The hardware will not work without the software and the software will not work without the hardware. That is what makes each of them a *component* of a patented invention. Both components are necessary—the installation of the software component onto the hardware component is what creates the infringing device. Moreover, executable software code can substitute for, and have the same technical effect as, specialized hardware in an electronic device. Installing software or firmware onto hardware "creates a new machine, because a general purpose computer in effect becomes a special purpose computer once it is programmed to perform particular functions pursuant to

---

[4] *See* Microsoft OEM System Builder License ¶ 7, http://oem.microsoft.com/downloads/Public/sblicense/English_SB_License.pdf (last visited Jan. 22, 2007).

[5] Certificate of Authenticity, http://oem.microsoft.com/script/sites/public/certificate_of_authenticity.htm (last visited Jan. 22, 2007) (emphasis in original).

instructions from program software." *In re Alappat*, 33 F.3d 1526, 1545 (Fed. Cir. 1994). When executable software code provides the same technical effect as specialized hardware, the code is just as much a "component" of an infringing device as the specialized hardware itself was in the past.

This Court's *Grokster* decision further supports treating hardware and software interchangeably. Drawing extensively from patent law decisions involving hardware devices, this Court found copyright liability for distribution of a device "with the object of promoting its use to infringe copyright." *MGM Studios, Inc. v. Grokster, Ltd.*, 125 S. Ct. 2764, 2780 (2005). However, the "device" in *Grokster* was software, not hardware. *Id.* at 2782 ("the inducement theory of course requires evidence of actual infringement by recipients of the device, the software in this case").

In *Grokster*, this Court did not hesitate to apply traditional principles of patent law to a new mechanism for copyright infringement, even though it was implemented by software and not by hardware. If this Court will adopt principles of patent law and apply them to software in copyright cases, it should apply the same principles to software in an actual patent case.

Against the background of the convergence of hardware and software, Microsoft and its amici raise two main arguments in favor of creating an exclusion for software in Section 271(f): that executable software code is actually mere "design information" and that software must be fixed in a particular physical medium to be a component.

As discussed previously, the primary output of new economy software companies is intangible intellectual

property. This is what they protect with their own software patents. It is incorrect for Microsoft and its amici to continually refer to software code as mere "design information," similar to a mold, blueprint, or specification. The executable code of Windows, for example, is not just an abstract idea or design information, it is functioning software that actually operates the computer—that is why it is called the computer's *operating system*.

This is very different from the use of a mold or prototype of a physical component. Creation of a component based on a mold or specification requires the use of additional materials to create a replica of the component. The mold itself is nonfunctional—it is only used to guide the manufacture of actual functional components. In contrast, the golden master disk contains the *actual executable Microsoft Windows software code*, which Microsoft intends to have installed on each computer. The executable software code is *not* design information. It is not a recipe or set of instructions for assembling a computer, or a blueprint for building a hard drive, or the CAD/CAM specifications for building a machine. The executable code is itself a component that is combined with the hardware to create each infringing computer. Whether or not the golden master itself, or a duplicate of the golden master, or a stream of bits sent over the digital transmission wire is used to install the software, the executable software code is the same.

Similarly, the executable software code is a component irrespective of its packaging. Just as gears and chips are supplied in customary packaging when shipped to the customer, software is likewise packaged in various physical media. For example, the installation CD is just such

a package for transporting the executable software code—once the software code is installed, the disk is removed. A permanent physical union of an installation disk and a computer never occurs when installing software. Imagine a customer buys a copy of Microsoft Word at the local computer store. When the customer takes the disk home and installs the software, the disk itself does not become part of the computer; it is removed when the installation is complete and the software code has been transferred. No "molecules" from Microsoft's headquarters merge with the user's personal computer. *See* Amazon Brief at 3. Yet it is clear that the executable software code on the computer is the code that was written and compiled by Microsoft.

The same would be true if the user downloaded the software over the Internet and did not purchase a CD at all. The intellectual property created by new economy companies "may be shipped on a disk or other physical product," but that is not necessary, and "software is increasingly shipped to the purchaser over the Internet." Landes & Posner at 390. If the code were distributed over the Internet, that would obviously not turn the Internet into a "component" of the patented invention either.

B. A Software Component Is Supplied When It Is Furnished To A User In An Authorized And Customary Manner

Once it is clear that the executable software code can be a component of a patented invention, the proper inquiry is to determine whether the code was supplied to a user "in such manner as to actively induce the combination" of the executable software code with other components outside of

the United States "in a manner that would infringe the patent if such combination occurred within the United States." 35 U.S.C. § 271(f)(1) (2007).

Whether a component is "supplied" must be viewed in context. The properties of a component may dictate how it is supplied to customers. For example, liquid components may be supplied in containers, or perhaps through pipelines. In each case, the most relevant inquiry is to look at the customary methods by which a seller furnishes a component to a buyer in that industry.

In the new economy, intangible intellectual property products, such as executable software code, are often furnished by providing one physical copy of the product (or an Internet download) along with a license to install that product on more than one machine. The number of products "supplied" is the number of products authorized or licensed for installation. A software *license*, not possession of a physical disk, is what gives a user the right to run a program or transfer it to others.[6] For example, a company with five or more employees can purchase software from Microsoft under a Volume License Agreement. The general license terms for Microsoft Windows XP Professional allow the company to "store one copy of the software" on a network server and

[6] *See* How to Buy Overview, http://www.microsoft.com/licensing/options.mspx (last visited Jan. 22, 2007).

"*use that copy to install the software on any other device to which a license has been assigned.*"[7]

If a software company furnishes a business with *one* software CD and permission to install that software on *five hundred* computers via the network, that software company has supplied its customer with executable software code for five hundred computers, not one. In fact, the software has been supplied in the most efficient, customary way. Neither party wants the software company to supply the executable code on five hundred separate CDs. As Microsoft points out, "*authorized* copying is inherent in the distribution of software." Microsoft Br. at 17. As a copyright holder, Microsoft has the exclusive right to prevent copying. *Id.* at 19 n.4. But when Microsoft *furnishes* the executable software code, *intends* for it to be copied, *allows* it to be copied, and gets *paid* for the copies, then Microsoft *supplied* those copies of the executable software code.

Furthermore, the number of physical disks and secondary copies involved in the installations is irrelevant to the users of the software. Does anyone who buys a computer with an official "Microsoft Windows" certificate of authenticity have *any* doubt that Microsoft supplied the executable software code? Microsoft's instructions to OEMs on how to install Windows illustrate that a physical disk is not necessary, or even desired, for the installation of the Windows operating system. Microsoft describes four

---

[7] Microsoft Desktop Operating Systems, Section A(II)(a), http://www.microsoftvolumelicensing.com/userights/ProductPage.aspx?pid=91 (click hyperlink which states "Standard Terms for Windows XP Professional, when licensed in a Microsoft Desktop Operating Systems model") (last visited Jan. 22, 2007) (emphasis added).

different ways OEMs can preinstall Windows onto personal computers. Microsoft, *Preinstalling Microsoft Windows XP Professional by Using the OEM Preinstallation Kit, Part I*, 2003, at 3. The recommended way is to fully install the software on one computer and then take an "image" of that first computer and install the image onto all of the other computers over the network. *Id.* According to Microsoft, this method is "faster and less cumbersome than the other models." *Id.* In fact, using the CD to manually install the software is the lowest-rated option because it is "slow and inefficient." *Id.* at 5 & 7.

Thus, new economy software companies like Microsoft rely on the fact that software can be supplied in many different ways, most of which are more efficient than manually using a separate CD for each computer. As the panel majority pointed out in the decision currently on appeal, "Microsoft has taken full advantage of the replicable nature of software to efficiently distribute Windows™ internationally. At the same time, however, Microsoft posits that § 271(f) liability should attach only to *each disk* that is shipped and incorporated into a foreign-assembled computer." *AT&T Corp. v. Microsoft Corp.*, 414 F.3d 1366, 1370 (Fed. Cir. 2005) (emphasis added). But the customary ways in which software is supplied do *not* require a separate disk for each installation—the executable software or firmware code for many machines can be furnished on one physical medium, such as a CD, or on none at all.

If Microsoft did provide OEMs with multiple operating system CDs and the OEMs installed each CD manually, then there is no doubt that Section 271(f) would apply. Microsoft conceded that fact in the district court.

AT&T Corp. v. Microsoft Corp., No. 01 Civ. 4872 (WHP), 2004 WL 406640, at *7 n.7 (S.D.N.Y. Mar. 5, 2004). The result should be the same if Microsoft provides the executable software component on one golden master CD and *authorizes, licenses, and instructs* the OEMs to install it on multiple computers. Whether liability attaches under Section 271(f) should not depend on which installation method the OEM chooses.

III. Liability For A Single Act In The United States Properly Encompasses All Consequential Damages, Including Foreign Sales

Lost in the pages of argument over convoluted interpretations of liability is the real issue: the proper measure of damages. Once a company supplies software components to foreign manufacturers in the same way that it supplies them to U.S. manufacturers, proper damages include foreign sales.

When a software company supplies executable software code components—even with a single act of sending a golden master disk or digital transmission—it should be held liable for all attendant damages. Software companies like Microsoft supply golden master disks with the knowledge and intent that the executable software code on those disks will be installed onto multiple hardware devices and sold overseas. Thus, those companies should pay damages for every infringing device that would not have been sold abroad but for their acts of supplying.

Microsoft and its amici contend that they will face endless liability if this Court does not create an exception for software in Section 271(f). But software companies do *not* face endless liability for the software components they have supplied—they only face liability for the number of copies they intend to be installed. This is the same liability they would face if the installation had occurred within the United States. For example, if Microsoft is liable for "tens of millions of foreign-produced" copies of Windows (Microsoft Br. at 24), that is only because it authorized and licensed the foreign OEMs to install Windows on tens of millions of computers, for which Microsoft will receive royalty payments.[8]

This outcome reflects the realities of software distribution in the real world. Even if a software company only sends one golden master CD abroad, foreign OEMs do not pay for only one copy of the software, as would a regular consumer purchasing a single copy of a software product. The OEMs pay for the executable software code to install on thousands of infringing devices. Thus, the appropriate measure of damages is not just for a single sale of one copy of the software. If a software company like Microsoft gets paid for each and every infringing computer that is created from that one golden master disk, it must also pay the consequential damages that flow from supplying that disk.

---

[8] In fact, as discussed *infra* in Section IV, if a new economy software company tactically encourages pirated copies of its software to be created and distributed so that it can establish or maintain a network monopoly, it can be liable for those copies as well. Even in that case, however, the company's liability is still limited by its intent.

Both patent law and copyright law already allow a plaintiff to collect damages for all of the consequences that flow from an infringer's actions, including foreign sales. Patent law allows a plaintiff to collect damages for foreign sales when infringing devices are made in the United States and sold abroad. *See Datascope Corp. v. SMEC, Inc*., 879 F.2d 820, 827 (Fed. Cir. 1989); *Schneider AG v. SciMed Life Sys. Inc.*, No. 94-1317, 1995 U.S. App. LEXIS 9754, at *9 (Fed. Cir. Apr. 26, 1995) (nonprecedential opinion). Copyright holders can also collect damages for foreign copies when the master copy was created in the United States. *See Update Art, Inc. v. Modiin Publ'g, Ltd.*, 843 F.2d 67 (2d Cir. 1988); *Sheldon v. Metro-Goldwyn Pictures Corp*., 106 F.2d 45, 52 (2d Cir. 1939).

Microsoft and its amici argue that applying the proper measure of damages would extend the extraterritorial reach of U.S. patent laws because the damages must be calculated by looking at the number of sales made abroad. *See, e.g.*, Br. of Intellectual Property Professors at 6-7. But "[a] claim is not 'extraterritorial' simply because it involves foreign acts or parties" and courts "may be too quick to perceive 'extra'-territoriality in claims that in fact allege multiterritorial infringements [that] involve acts or parties located in more than one country." Jane C. Ginsburg, *Extraterritoriality and Multiterritoriality in Copyright Infringement*, 37 Va. J. Int'l L. 587, 588 (1997).

The executable software code is developed in the United States and sent overseas for installation with authorization from the United States software company. Requiring the infringer to pay damages for such a

multiterritorial act, which has its origin in the United States, does not extend the extraterritorial reach of U.S. patent law.

Microsoft and its amici also argue that the entire software industry will move its research and development facilities overseas if they have to pay damages on foreign sales under Section 271(f). This scenario is implausible at best.

Because their primary output is intellectual property, new economy software companies benefit from strong intellectual property protection and are not likely to move to countries with weaker protection just to avoid application of Section 271(f). The patentability of software has been a contentious issue in Europe. It is very hard to believe that a U.S. software company would move to a country where it might not be able to patent its products at all, just to avoid one section of the U.S. patent law. It is even less likely that software companies will move to countries with less-developed intellectual property regimes, such as China.

New economy software companies could achieve cost savings by moving their development and production facilities overseas, but they have chosen to remain in the United States and have flourished here, in no small part because the copyright, trade secret, and judicial processes in the United States provide strong and effective protection for the intellectual property content of software products. There is no justification for letting such companies enjoy the benefits of our strong intellectual property system for their own products while, at the same time, allowing them to avoid paying damages for infringing other companies' patents when those same products are exported.

IV. Software Companies Advocate Weak Protection Overseas Because Software Piracy In Developing Countries May Be Beneficial In The New Economy

Given the important role that patents play in encouraging innovation, it may seem surprising that software companies are arguing for *less* patent protection for software. Why would companies with numerous software patents argue for an exemption that would also result in less patent protection in countries with weaker intellectual property rights? The answer lies in the economic nature of new economy software markets, particularly the monopolies that exist in these markets.

Software markets exhibit a unique combination of economic factors. First, intellectual property is characterized by high fixed costs relative to marginal costs; for computer software the marginal cost approaches zero. Richard A. Posner, *Antitrust in the New Economy*, 68 Antitrust L.J. 925, 926-27 (2001). In fact, the marginal cost can actually be negative. Richard A. Posner, *The Law & Economics of Intellectual Property*, 131:2 Daedalus 5 (2002) ("Indeed, widespread use of intellectual property can actually increase the value of the property; in effect, additional copies have negative cost, when the value they confer is taken into account.").

Second, many computer software products, particularly operating systems and design products like AutoCAD, are characterized by network effects. Network effects occur in markets where the larger the firm's output,

the more valuable that output is to its customers. Posner, *Antitrust in the New Economy*, at 928. Computer software products become more valuable as more people use them, because of training, support, and standardization of equipment and procedures. *Id.*

Third, intellectual property is a public good. A public good is something that one person can use without reducing the value of the product to other users. Richard A. Posner, *Do We Have Too Many Intellectual Property Rights?*, 9 Marq. Intell. Prop. L. Rev. 173, 178-79 (2005). Thus, "piracy" of intellectual property is not really like theft of physical property. *Id.* If someone makes a pirated copy of the Windows operating system, for example, the program is still there for Microsoft to sell to others. *Id.* at 179. If the software pirate could not have afforded to purchase the software, then Microsoft has not actually lost any sales. *Id.* In fact, if only a few of the people who copy an operating system would have bought it, Microsoft benefits from the piracy because the more people who use the operating system, the more people who need applications for it. *Id.*

The interplay of these economic factors explains why software companies like Microsoft are willing to tolerate, or perhaps even encourage, software piracy. As Landes & Posner point out, "[e]ven pirating of software, about which the software industry complains so loudly, is not all loss to software producers." Landes & Posner, *supra*, at 46.

In other words, the existence of pirated software increases demand for complementary products. *Id.* For example, the proliferation of pirated copies of Windows may benefit Microsoft and other software companies who write

applications that run on the Windows platform. In addition, "[b]y accelerating the spread of the work" to more users "piracy may help the creator of the work to obtain a network monopoly." *Id.* Software piracy can actually *help* the producer of the pirated software by increasing the network externalities of the dominant firm. *Id.* at 393.

Since Microsoft and its amici do not need the protection of Section 271(f), they would like this Court to create an exception for the entire software industry. If this Court does so, it will improperly favor new economy software companies over traditional electronics companies who rely on the protection of Section 271(f) to recoup their research and development costs.

CONCLUSION

The judgment below should be affirmed.

Respectfully submitted,

Jack E. Haken
Edward Blocker
PHILIPS INTELLECTUAL PROPERTY & STANDARDS
345 Scarborough Road
Briarcliff Manor, NY 10510

John M. DiMatteo
Counsel of Record
Eugene Chang
WILLKIE FARR & GALLAGHER LLP
787 Seventh Avenue
New York, NY 10019

Counsel for Amici Curiae U.S. Philips Corporation and Philips Electronics North America Corporation

January 23, 2007