NOTES

DOES STRICT TERRITORIALITY
TOLL THE END OF SOFTWARE PATENTS?

James Ernstmeyer[*]

INTRODUCTION

Suppose that an American inventor develops a software application that translates audio files of German speech into files of German text on a personal computer. The inventor has obtained a United States patent, hoping to recoup the expense of developing the software by excluding others from making or

selling it. The principal market for this software is in Europe, and the inventor counts on exporting the software into a lucrative European market. The European market turns out to be lucrative indeed. Unfortunately, the inventor does not reap the benefit because, in spite of the patent, a giant United States software manufacturer has incorporated the novel software in its popular operating system and sent it to Germany, where it is loaded on personal computers that are sold throughout the European Union.

The inventor is disappointed with the protection of the U.S. patent. The combined effects of recent holdings by the Supreme Court and the Court of Appeals for the Federal Circuit leave the inventor no remedy under U.S. patent law against a domestic competitor who sells infringing software overseas. The inventor instead must apply for a patent in every foreign country where a competitor might manufacture or sell computers containing the software. Software is not patentable in all countries. Even where it is patentable, the inventor still faces the challenge of obtaining a remedy – often available only in the impractical form of damages from individual consumers.

The inventor chose to invest time and money in patenting the software, even though copyright protected the software from the time it was committed to paper or stored on a hard drive. A patent seemed attractive because it would protect the functional elements of software, not just the expressive elements that copyright protects. Furthermore, a patent suit against an alleged infringer would sidestep the potential defense that the software was created independently, which is an absolute defense under copyright but is no defense under patent law. As a result of the recent decisions, patent protection and its advantages are no longer available for software developed in the United States and sold overseas, but copyright continues to provide protection, and even some offsetting advantages, to both the inventor and the public.

The overseas market continues to gain economic importance, and in some cases is the primary market for an invention. As a result, the international trade in software licenses provides great and growing value to the U.S. economy. As license revenue is available only if software can be protected as intellectual property, high stakes ride on whether software is patentable. Unfortunately, recent judicial developments have rendered patent protection ineffective for software in cross-border trade. These developments suggest that American software inventors could save the expense of obtaining patents and instead enforce their copyrights to restrict unauthorized sales overseas more effectively.

This Note does not consider trade secrets, the only other means to protect intellectual property rights in software. Instead, the focus is on patents and copyrights, which place no burden of secrecy on the inventor and thereby leave open the opportunity for the public to build upon the software's innovative

content. Moreover, it is unclear that trade secrecy is even capable of protecting property rights in publicly distributed software.[1]

Part I traces the brief history of intellectual property protection for software-based inventions, examining the benefits and detriments of both copyright and patent protection as they pertain to software. In particular, it examines the extent to which the recent Federal Circuit decision *In re Bilski*[2] casts doubt on the patentability of software. Part II discusses the economic importance of intellectual property rights in software in cross-border trade. It also explains how those rights are jeopardized by the Supreme Court decision in *Microsoft v. AT&T*,[3] which opened a gap in the patent protection for devices containing software, allowing manufacturers to evade penalties for infringement simply by sending software and hardware separately to an overseas location where it is then assembled into a patented device. Part III argues that the combined effect of *In re Bilski* and *Microsoft v. AT&T* is to eliminate practical patent protection for exports of software-based inventions. In light of these decisions, this Note proposes that software innovators spare the expense of obtaining patents and turn instead to litigating their copyrights under both United States and international law to achieve more reliable protection of their intellectual property in the global economy.

I.   UNITED STATES PROTECTION FOR INTELLECTUAL PROPERTY RIGHTS IN SOFTWARE

A.   *Defining Software*

At the outset, it is necessary to define software in a way that covers the many forms that software can take. Electronic products, from communications satellites to talking greeting cards, incorporate software expressed in tangible forms that range from hand-written source code to micro-fabricated, permanently-wired, transistor arrays.[4] Software need not even take a tangible

---

[1] *See, e.g.*, Young Dental Mfg. Co. v. Q3 Special Prods., Inc. 891 F. Supp. 1345, 1350 (E.D. Mo. 1995) (rejecting any suggestion that "publicly sold software" could be the subject of a trade secret); Mark A. Lemley & David McGowan, *Legal Implications of Network Economic Effects*, 86 CAL. L. REV. 479, 528 n.211 (1998) (noting that Microsoft's assertion of trade secret protection for its widely distributed software "flies in the face of the fundamental tenets of trade secret law").

[2] 545 F.3d 943 (Fed. Cir. 2008), *cert. granted sub nom.* Bilski v. Doll, 129 S. Ct. 2735 (June 1, 2009) (No. 08-964).

[3] Microsoft Corp. v. AT&T Corp., 550 U.S. 437 (2007).

[4] *See* Glynn S. Lunney, Jr., Note, *Copyright Protection for ASIC Gate Configurations: PLDs, Custom and Semicustom Chips*, 42 STAN. L. REV. 163, 164-65 (1989) (describing how courts have struggled with the scope of copyright protection for computer code manifested in different forms of hardware).

form, and can instead exist in the abstract as a set of instructions not tied to any medium.[5]

The Copyright Act defines software as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result."[6]   The Supreme Court recently embraced an equivalent, functional definition of software as a "set of instructions . . . that directs a computer to perform specified functions or operations."[7]   The definition of software as a set of instructions is broad enough to include both software in the abstract and software in tangible forms.

B.   *Copyright Protection for Software*

In 1974, when widespread use of personal computers was still on the distant horizon,[8] Congress foresaw that software would create unprecedented challenges for intellectual property law.   To address those challenges, it established the National Commission on New Technological Uses of Copyrighted Works.[9]   The Commission recommended that Congress amend the copyright law to explicitly recognize computer programs as the "proper subject matter of copyright."[10]   In response to this recommendation, Congress in 1980 rewrote a provision of the Copyright Act of 1976 that had previously guaranteed copyright protection for creative works stored or processed on computers to protect the computer programs themselves instead.[11]   Courts

---

[5] *Microsoft v. AT&T*, 550 U.S. at 447-48 (analogizing software in the abstract as the notes of a Beethoven symphony and tangible software as the sheet music for the same symphony).

[6] 17 U.S.C. § 101 (2006).   The statute defines a "computer program," a term that is synonymous with "software."   *See* Parker v. Flook, 437 U.S. 584, 587 n.7 (1978).

[7] *Microsoft v. AT&T*, 550 U.S. at 447.

[8] IBM Archives: IBM Personal Computer, http://www.ibm.com/ibm/history/exhibits/pc/pc_1.html (last visited May 6, 2009) (chronicling the introduction of the first IBM personal computer in 1981).

[9] Pub. L. No. 93-573, § 201, 88 Stat. 1873, 1873 (1974) (creating a commission to study the "use of copyrighted works of authorship . . . in conjunction with automatic systems capable of storing, processing, retrieving, and transferring information").

[10] NAT'L COMM'N ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, FINAL REPORT OF THE NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS 1 (1978), http://digital-law-online.info/CONTU/PDF/Chapter1.pdf.

[11] *Compare* Pub. L. No. 94-553, § 117, 90 Stat. 2541, 2565 (1976) (affording to copyright owners the same "rights with respect to the use of the work in conjunction with automatic systems" as otherwise), *with* Pub. L. No. 96-517, § 10, 94 Stat. 3015, 3028-29 (1980) (defining "computer program" and limiting software copyright owner's exclusive right to reproduce and adapt the software when copying or adapting is essential to using the program), *and* H.R. REP. NO. 96-1307(I), at 23 (1980), *reprinted in* 1980 U.S.C.C.A.N. 6460, 6482 (acknowledging that the 1980 amendment "embodies the recommendations of the Commission on New Technological Uses of Copyrighted Works with respect to clarifying the law of copyright of computer software").

applied the new law in the early 1980s to protect software from direct, literal copying.[12] All software – even software in a form unreadable by humans – received the same protection against literal copying as Ernest Hemingway's *For Whom the Bell Tolls*.[13]

However, protection against literal copying is but a narrow slice of the copyright protection afforded to literary works, which are also protected against "substantial" copying, even if it is not word-for-word.[14] Protection against non-literal copying is feasible only if the courts can distinguish elements of creative expression, which are protectable, from the overarching ideas of a work, which are not.[15] This task is more difficult for software because fine details that would count as creative expression in a traditional literary work can instead encode the overarching idea of a piece of software.[16]

By the 1990s, courts had developed a systematic method to separate software into protectable expressive elements and unprotectable idea elements.[17] The method classifies elements of software in categories that vary by degree of abstraction.[18] The most abstract elements, including the purpose of the software and its overall function, are always ideas and therefore unprotectable by copyright.[19] Least abstract are the literal text of the software in human-readable form and, ultimately, the object code that directs the actions of the computer.[20] These elements are always expressive and therefore subject

---

[12] Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1247-48 (3d Cir. 1983) (concluding that computer programs are copyrightable based on the 1980 amendment to the Copyright Act).

[13] *Id.* at 1249 (holding that a piece of source or object code, while not a Hemingway novel, was nevertheless protected against unauthorized copying as a literary work).

[14] Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930) (opining that copyright "cannot be limited literally to the text, else a plagiarist would escape by immaterial variations").

[15] *Id.* (finding that copyright law should not allow an author to "prevent use of his 'ideas,' to which, apart from their expression, his property is never extended").

[16] *See id.* (acknowledging the difficulty of finding a boundary between idea and expression in the context of a literary work).

[17] Computer Assocs. Int'l, Inc. v. Altai, Inc., 982 F.2d 693, 706-11 (2d Cir. 1992) (articulating the "abstraction-filtration-comparison" test to separate idea from expression in computer programs).

[18] Gates Rubber Co. v. Bando Chem. Indus., Ltd., 9 F.3d 823, 835 (10th Cir. 1993).

[19] *Id.* at 836 ("For example, the main purpose or function of a program will always be an unprotectable idea. . . . Likewise, . . . the basic function or purpose of a module will nearly always be an unprotectable idea or process.").

[20] *Id.* at 835 ("Source code is the literal text of a program's instructions written in a particular programming language. . . . Object code is the literal text of a computer program written in a binary language through which the computer directly receives its instructions.").

to copyright protection.[21] In between, "[t]he intermediate levels of abstraction, such as structure, sequence, organization, and the like, are less prone to generalizations."[22] Within those intermediate levels the court must draw the line between protectable expression and unprotectable idea depending on the facts of each case.[23] No matter where the court draws that line, this method enables copyright in software to protect against more than the strictly literal duplication of the source code.

C. *Patent Protection for Software*

Patent protection for software has been slower to develop and more controversial to apply than copyright protection. The question of software patentability hinges on whether processes and mathematical algorithms are patentable. Ultimately, any computer program implements an algorithm, defined as a "procedure for solving a given type of mathematical problem."[24] Such a procedure may in turn qualify as a patentable process if it "represent[s] the means or method of producing a result."[25] Software expressing an algorithm is thus potentially subject to patent as a "new and useful process."[26] Whether a particular computer program could be patented was initially a close question for two reasons. First, the software might implement an algorithm that simply restates a law of nature, so that to patent the software is tantamount to patenting the law of nature itself, which is an impermissible outcome.[27] Second, software operating within a digital computer may not produce a "result" within the meaning of prior cases from before the digital era, when courts generally required patented processes to produce concrete physical results, such as "tanning, dyeing, making water-proof cloth, vulcanizing India rubber, [and] smelting ores."[28]

---

[21] *Id.* at 836 ("At the other end of the abstractions spectrum, source and object code, which are the literal elements of a program, will almost always be found to be protectable expression unless the doctrines of merger and *scenes a faire* come into play.").

[22] *Id.*

[23] *Id.* (giving examples of when "the structure of one program may be unprotectable because it constitutes an idea, [but] the organization and arrangement of another program may be expressive and thereby protectable").

[24] Gottschalk v. Benson, 409 U.S. 63, 65 (1972).

[25] Diamond v. Diehr, 450 U.S. 175, 182 (1980) (quoting Corning v. Burden, 56 U.S. (15 How.) 252, 267-68 (1853)).

[26] 35 U.S.C. § 101 (2006) ("Whoever invents or discovers any new and useful process, machine, manufacture, or composition of matter, or any new and useful improvement thereof, may obtain a patent therefor . . . .").

[27] *Benson*, 409 U.S. at 71-72 (rejecting a patent for a computer program that converted digital numbers into numerical representations because a patent on an algorithm used exclusively within a general purpose computer would be too broad, effectively pre-empting all uses of the underlying mathematical formulation itself).

[28] *Diehr*, 450 U.S. at 182-83.

Software in the abstract ("intangible" software embodying a mathematical algorithm) was initially found to be equivalent to an abstract idea or a law of nature, which by itself failed to produce any physical transformation, and therefore was not patentable subject matter.[29] Nevertheless, patent protection was available for software embedded in a patentable device or process that did produce a physical result.[30] The combined effect of these holdings was a "mathematical algorithm exception," under which a mathematical algorithm is unpatentable unless it is applied to produce a tangible result.[31] In refining the mathematical algorithm exception, the Federal Circuit has determined that an algorithm can produce a tangible result if it transforms digital data into a smooth curve displayed on a computer monitor.[32] Likewise, an algorithm that transforms a heart patient's digitized electrocardiograph signal into another electrical signal representing the condition of his heart also produces a tangible result.[33] The algorithm that displayed the curve on the monitor was expressed as an electrical circuit dedicated to that purpose, while the one that transformed the heart signal was expressed as software on a digital computer.[34] Whether or not software mediated the process, the court required that the algorithm be used as a means to produce a physical result – a curve on a monitor in one case and an electrical signal in the other.

In 1998, *State Street Bank & Trust Co. v. Signature Financial Group, Inc.*,[35] severed the connection between the algorithm and a physical result, holding that software running on a general purpose computer could itself be an application of a mathematical algorithm.[36] The transformation of financial

---

[29] *Benson*, 409 U.S. at 70-72 ("Transformation and reduction of an article 'to a different state or thing' is the clue to the patentability of a process claim that does not include particular machines.").

[30] *Diehr*, 450 U.S. at 187 (reiterating that "an application of a law of nature or mathematical formula to a known structure or process may well be deserving of patent protection").

[31] State St. Bank & Trust Co. v. Signature Fin. Group, Inc., 149 F.3d 1368, 1373 n.4 (Fed. Cir. 1998), *abrogated in part by In re* Bilski, 545 F.3d 943, 960 n.19 (Fed. Cir. 2008), *cert. granted sub nom.* Bilski v. Doll, 129 S. Ct. 2735 (June 1, 2009) (No. 08-964).

[32] *In re* Alappat, 33 F.3d 1526, 1544 (Fed. Cir. 1994) (holding that electronic circuitry used to create a smooth display is eligible for patent because it "is not a disembodied mathematical concept which may be characterized as an 'abstract idea,' but rather a specific machine to produce a useful, concrete, and tangible result").

[33] Arrhythmia Research Tech., Inc. v. Corazonix Corp., 958 F.2d 1053, 1059 (Fed. Cir. 1992) (holding patentable a "method for analyzing electrocardiograph signals" because signals represent steps in a physical process).

[34] *Alappat*, 33 F.3d at 1538-39 (describing the discrete electronic devices used to implement the algorithm); *Arrhythmia Research*, 958 F.2d at 1055 ("Certain steps of the invention are described as conducted with the aid of a digital computer . . . .").

[35] 149 F.3d 1368.

[36] *Id.* at 1373 (holding that software running on a general purpose computer is patentable subject matter if it "produces 'a useful, concrete and tangible result'" and distinguishing

data into the share price of a mutual fund through use of software produced a "useful, concrete and tangible result" for purposes of patentability.[37]   By recognizing patentable subject matter in an invention that did not produce a physical result, this decision opened the door to patents on software in the abstract.

D.   *The Backlash Against Intellectual Property Rights in Software*

Even as courts have arrived at workable means to protect intellectual property rights in software, the very idea of property rights in software has generated controversy. Numerous commentators have pointed out that such property rights threaten many of the social benefits of software-based technology.[38] Chief among the jeopardized benefits is peer production, the informal production of information by teams of peers, which rivals traditional production by hierarchical organizations.[39] Wikipedia volunteers can create an online encyclopedia comparable to the *Columbia Encyclopedia*, and volunteer "clickworkers" can collaboratively map the lunar surface for NASA.[40]

Proponents of a philosophy known as "open source software" tout the higher quality and innovativeness of software that is peer-produced by non-hierarchical, collaborative methods.[41] Software development on the open source model entails large numbers of self-selected developers collaborating to perfect a single piece of software without any external organizing force and without any interest in owning the final product.[42]

---

from *Gottschalk v. Benson*, 409 U.S. 63, 71-72 (1972), in which the claimed algorithm had no application other than on a computer and therefore a patent on the software would pre-empt all uses of the algorithm (quoting *Alappat*, 33 F.3d at 1544)).

[37] *Id.* at 1370-73 (concluding that "a final share price momentarily fixed for recording and reporting purposes" constitutes a tangible result).

[38] *E.g.*, Lawrence Lessig, *The Death of Cyberspace*, 57 WASH. & LEE L. REV. 337, 344 (2000) (contending that software copyright will stamp out the flowering of creativity fertilized by the internet).

[39] Yochai Benkler, *Coase's Penguin, or, Linux and the Nature of the Firm*, 112 YALE L.J. 369, 383-87 (2002) (explaining the growth of nonprofessional information production as a consequence of the low transaction costs afforded by the internet commons).

[40] *Id.* (citing successful examples of online peer production).

[41] Ronald J. Mann, *Commercializing Open Source Software: Do Property Rights Still Matter?*, 20 HARV. J.L. & TECH. 1, 21-22 (2006) (acknowledging that "widespread adoption of commercially successful open source products offers strong testimony that in some contexts the collaborative development model can produce software of high quality and easy interoperability"); Robert P. Merges, *A New Dynamism in the Public Domain*, 71 U. CHI. L. REV. 183, 192-93 (2004) (observing that even mainstream software giants like IBM have embraced open source software as a means of avoiding costly intellectual property entanglements with Microsoft operating systems).

[42] Benkler, *supra* note 39, at 381-82 (comparing Open Source to other instances of peer production).

Open source adherents regard strong intellectual property rights in software as a threat because strong ownership rights impede the rapid evolution of software into a highly efficient and reliable product.[43] Moreover, software intellectual property is widely believed to undermine collaborative software development by raising the low transaction costs on which the approach is based.[44] Others, in the "free software movement," object to software IP on the purely moral ground that it unjustly limits the freedom of users by forcing them to submit to restrictions on the modification and reuse of software.[45]

The preceding objections to intellectual property rights in software apply to both patents and copyrights. However, patents are more troublesome because of the broad scope of the rights they confer and the strict liability they impose for infringement. The Federal Circuit has permitted software patentees to claim a right to the broad functions of their software not tied to any particular implementation.[46] This expansive right contrasts with the constrained scope of a copyright in software, which is confined to the expressive elements of the software text itself. Copyright owners can thus prevent a smaller range of disseminating activities than patentees. They can even use their copyright in connection with a free software license as a tool to promote the free redistribution and modification of their software.[47]

---

[43] *See* Open Source Initiative, The Open Source Definition (Annotated), http://opensource.org/docs/definition.php (last visited Aug. 10, 2009) (promoting software license terms that "make evolution easy" by permitting access to source code and allowing users to modify and redistribute software).

[44] Benkler, *supra* note 39, at 445 (decrying strengthened intellectual property protection that threatens to destroy a "robust peer production sector").

[45] Richard Stallman, Why "Open Source" Misses the Point of Free Software, http://www.gnu.org/philosophy/open-source-misses-the-point.html (June 22, 2009) (arguing that "software can only be said to serve its users if it respects their freedom"). The free software movement distinguishes itself from the open software movement on the basis of its differing philosophical motivation. *Id.* ("We [of the free software movement] disagree with the open source camp on the basic goals and values . . . ."). Free software adherents, who explain themselves with the slogan "Think of free speech, not free beer," emphasize that it is unethical to restrict users from studying, changing, and redistributing the software they own or license. *Id.* (describing the ethical imperative to "avoid the non-free operating systems that deny freedom to their users"). Open source adherents object to the same restrictions, but on a practical, not an ethical, basis. *Id.* ("The idea of open source is that allowing users to change and redistribute the software will make it more powerful and reliable.").

[46] JAMES BESSEN & MICHAEL J. MEURER, PATENT FAILURE: HOW JUDGES, BUREAUCRATS, AND LAWYERS PUT INNOVATORS AT RISK 210 (2008) ("Software patents do not, in general, need to include the computer code nor detailed flowcharts nor any other detailed description of specific operation.").

[47] Free Software Foundation, Inc., How to Use GNU Licenses for Your Own Software, http://www.gnu.org/licenses/gpl-howto.html (June 17, 2009) ("[I]t is a very good idea to register the copyright with the US Registry of Copyrights, because that puts you in a stronger position against anyone who violates the [free software] license in the US.").

A patent owner's property right is stronger for the added reason that an infringer is strictly liable for making, using, or selling a patented device or process.[48] By contrast, a finding of copyright infringement requires that the plaintiff prove both copying and misappropriation, subject to the defense that the defendant independently created the work.[49] The constrained rights and greater burden of proof of the copyright plaintiff both act to reduce the copyright threat to peer production.

On the other hand, developers of non-peer-produced software have embraced patents for the broad, strong property rights they provide. More than 170,000 software patents have been issued since *State Street Bank*, some covering code as commonplace as that which generates pop-up windows on an internet browser, igniting a firestorm of controversy on policy grounds.[50] Critics rail that by over-expanding the subject matter covered by patents, *State Street Bank* is the emblem for a "second enclosure movement," fencing off the intellectual commons from creative workers who depend on unfettered access to ideas for their livelihood.[51] In particular, an essential feature of software innovation is that software developers build incrementally upon existing blocks of code that perform functions that contribute to the software being developed.[52] This cumulative innovation process suffers when one developer receives a patent for a broadly-worded, abstract software function, thereby fencing many unanticipated software functions out of the intellectual commons.[53] Subsequent developers who need to license the already patented function are frustrated.[54] In this way, the often uncertain boundaries of the

---

[48] 35 U.S.C. § 271(a) (2006) ("[W]hoever without authority makes, uses, offers to sell, or sells any patented invention, within the United States or imports into the United States any patented invention during the term of the patent therefor, infringes the patent.").

[49] Arnstein v. Porter, 154 F.2d 464, 468 (2d Cir. 1946).

[50] BEN KLEMENS, MATH YOU CAN'T USE: PATENTS, COPYRIGHT, AND SOFTWARE 1-5 (2005) (arguing that software should not be patentable subject matter).

[51] James Boyle, *The Second Enclosure Movement and the Construction of the Public Domain*, 66 LAW & CONTEMP. PROBS. 33, 38, 48-51 (2003) (warning that a too strong intellectual property rights regime will diminish the public domain of ideas, destroy the promise of peer-to-peer innovation, and threaten the American system of science).

[52] *See* BESSEN & MEURER, *supra* note 46, at 8-9 (reporting that a single e-commerce software application could easily infringe more than ten thousand prior software inventions, illustrating the extent to which software innovation builds upon prior development).

[53] *Id.* at 198-203 (arguing that software patents are especially prone to creating disincentives for legitimate inventors because the abstract nature of software patent claims permits patentees to claim infringement by legitimate inventors of unforeseen inventions in unrelated fields).

[54] *Id.* (describing how AT&T got a software patent covering many widely used equivalent programming techniques and forced IBM, who practiced these techniques, into a cross-license).

property right conferred by a software patent impede the patent grant's goal of promoting innovation.[55]

The property fences created by software patents especially encroach on the free and open source software domain.[56] In order to coexist with the growing patent portfolios of traditional software developers, the open source movement has been forced to find its own form of intellectual property protection under the umbrella of large corporate sponsors.[57] In this alliance, large software and equipment developers benefit from multi-company, collaborative software development in the open source environment and in return promise to ward off patent infringement suits by outsiders against open source contributors.[58] In effect, large corporate stakeholders have erected a fence around pieces of open source territory, an unwelcome constraint from the standpoint of open source purists – to whom software patents are an anathema in the first place.

Other commentators view favorably the effects of large firm software patent portfolios on open source. For example, when IBM invests heavily in the open source Linux operating system, it brings private sector fortification to an open source domain that might otherwise be without a viable revenue source.[59] Focusing narrowly on the detriments of patent-based property rights in software also ignores the countervailing benefits that society gains by public disclosure of patented technologies. Without patent disclosure, such pivotal software developments as public-key cryptography might still be locked away in trade secrets instead of stimulating vital academic research.[60] Furthermore, while restoring a strict, pre-*State Street Bank* requirement for a connection between a software algorithm and a physical apparatus might help clarify the boundaries of property rights in patented software, it would also turn back the clock to a time when electromechanical devices predominated and leave radically new technologies unprotected.[61]

---

[55] *Id.* (explaining how software patents provide unclear notice to potential infringers more often than many other types of technology patents).

[56] 35 U.S.C. § 271 (2006) (granting the patentee the right to prevent others from making, using, selling, offering to sell, or importing the patented product or process); Free Software Foundation, Categories of Free and Non-Free Software, http://www.gnu.org/philosophy/ categories.html (June 22, 2009) (explaining that proprietary software and free/open source software are mutually exclusive categories).

[57] Mann, *supra* note 41, at 42-43 (describing contracts between open source developers and corporate sponsors to limit patent infringement claims).

[58] *Id.* at 30-31 (arguing that large software and equipment manufacturers, having come to depend on open source software to escape domination by Microsoft, defend their open source technologies from infringement suits as vigorously as their proprietary technology).

[59] Merges, *supra* note 41, at 192-93 (describing how IBM's investment in Linux helps to challenge competitor Microsoft's dominion over the PC operating platform).

[60] Martin Campbell-Kelly, *Not All Bad: An Historical Perspective on Software Patents*, 11 MICH. TELECOMM. & TECH. L. REV. 191, 229-30 (2005).

[61] Brief of Yahoo! Inc. and Professor Robert P. Merges as Amici Curiae Supporting Appellee at 11-12, *In re* Bilski, 264 F. App'x 896 (Fed. Cir. 2008) (No. 2007-1130)

E.   In re Bilski *Turns Back the Clock*

The patentability of software is the subject of vigorous ongoing debate not only in the academic community, but in the courts as well. In October 2008, the Federal Circuit reconsidered *State Street Bank* and the legal standard for the patentability of processes including software.[62] In an apparent throwback to pre-*State Street Bank* jurisprudence, *In re Bilski*[63] embraces a "machine-or-transformation test" for whether a process is patentable.[64] Under the test, patent claims incorporating a mathematical algorithm (as one kind of "fundamental principle") must connect that algorithm with a specific machine or specific physical transformation, and thereby ensure that subsequent innovators are free to use the algorithm in connection with different machines or transformations.[65]

Requiring a connection between the algorithm and a specific application rejects patent claims that would preempt a potentially wide range of applications not envisioned and disclosed by the inventor.[66] Bilski's method of calculating commodity transactions to hedge the risks of buyers and sellers was unpatentable because it incorporated a mathematical algorithm tied neither to a machine (not even a computer) nor to a transformation of any article, and thereby impermissibly preempted all applications of hedging in the field of consumable commodities trading.[67]

In re Bilski expressly reserved for future decisions the question of whether software is patentable subject matter as a category and whether the application of an algorithm on a computer constitutes a sufficient connection to a machine to render it patentable.[68] However, in abrogating the "useful, concrete and tangible result" test of *State Street Bank*, the Federal Circuit likely arrested the

---

(arguing for a flexible test under which a process would be eligible for patent if it is "machine-like," even if not machine-implemented).

[62] *In re* Bilski, 264 F. App'x 896, 897 (Fed. Cir. 2008) (considering "[w]hat standard should govern in determining whether a process is patent-eligible subject matter" and even whether to overrule *State St. Bank*).

[63] 545 F.3d 943 (Fed. Cir. 2008).

[64] *Id.* at 954-56 (holding a claimed process to be patent eligible "if: (1) it is tied to a particular machine or apparatus, or (2) it transforms a particular article into a different state or thing," recognizing a test of patentable subject matter previously announced by the Supreme Court in *Gottschalk v. Benson*, 409 U.S. 63 (1972)).

[65] *Id.* (arguing, for example, that "a claimed process involving a fundamental principle that uses a particular machine or apparatus would not pre-empt uses of the principle that do not also use the specified machine or apparatus in the manner claimed").

[66] *Id.* (describing concern about allowing inventors to monopolize a fundamental principle by pre-empting all uses of that principle).

[67] *Id.* at 964-66 (rejecting arguments that the claimed process is patentable on grounds that it "produces useful, concrete, and tangible results" and that it includes "physical steps").

[68] *Id.* at 961-62 ("We leave to future cases the elaboration of the precise contours of machine implementation, as well as the answers to particular questions, such as whether or when recitation of a computer suffices to tie a process claim to a particular machine.").

post-*State Street Bank* momentum toward broad patentability of intangible software.[69] The machine-or-transformation test is satisfied by the electronic transformation of data representing a specific physical object, without any transformation of the underlying physical object that the data represents.[70] For example, a qualifying process might detect and amplify the electrical signals that control a patient's heart without causing any change in the underlying signals themselves. This criterion virtually repeats the mathematical algorithm exception as applied by the Federal Circuit prior to *State Street Bank*, suggesting that the court has indeed turned back the clock to a date when a process incorporating software was patentable subject matter only when the process entailed a physical transformation.[71]

In one of three vigorous dissents, Judge Rader goes further and derides the majority opinion as a throwback to the Steel Age, fearing that the machine-or-transformation test will jeopardize the development of future technologies, such as software, by rendering them unpatentable.[72] Thus, while *In re Bilski* declines to adopt a per se rule stating that intangible software is not patentable, it nevertheless places a thumb – perhaps both thumbs – on the scale against patentability.

There is already evidence that lower courts and the Patent and Trademark Office's Board of Patent Appeals and Interferences are using the machine-or-transformation test to broadly invalidate software-related patents.[73] However, the *In re Bilski* Federal Circuit majority will not have the final word regarding the test of patentable subject matter. The Supreme Court has granted a petition of certiorari in the case.[74] While the patentability of software in general will not be before the Court, the Court's holding may nevertheless effectively decide the question.

---

[69] *Id.* at 960 n.19 ("[T]hose portions of our opinions in State Street . . . relying solely on a 'useful, concrete and tangible result' analysis should no longer be relied on.").

[70] *Id.* at 962-63 (concluding that the transformation of medical imagery data representing various types of bodily tissue into a "particular visual depiction of a physical object on a display" would satisfy the transformation branch of the test).

[71] *See supra* notes 32-34 and accompanying text (describing the test for patentability prior to *State Street*).

[72] *Bilski*, 545 F.3d at 1011-15 (Rader, J., dissenting) (arguing that the majority's decision harms future technologies by "link[ing] patent eligibility to the age of iron and steel at a time of subatomic particles and terabytes").

[73] CyberSource Corp. v. Retail Decisions, Inc., 620 F. Supp. 2d 1068, 1071, 1078 (N.D. Cal. 2009) (holding that a patent claim for "[a] computer readable medium containing program instructions for detecting fraud in a credit card transaction between a consumer and a merchant over the Internet" was invalid because it failed to meet either prong of the machine-or-transformation test); Petitioner's Reply Brief at 8, Bilski v. Doll, No. 08-964 (U.S. May 8, 2009) (citing two cases in which the U.S.P.T.O. Board of Patent Appeals and Interferences used the *In re Bilski* holding to invalidate software-related patents).

[74] Bilski v. Doll, 129 S. Ct. 2735 (June 1, 2009) (No 08-964).

II. UNITED STATES SOFTWARE PATENTS IN CROSS-BORDER TRADE

A. *Global Economic Stakes in United States Software Patents*

The large economic interests at stake in global high technology manufacturing and licensing magnify the importance of the debate over the patentability of software. Nations compete to attract high technology manufacturing, including software, because high technology industries yield greater value-added revenue and pay higher salaries than other types of manufacturing.[75] Software development and electronics manufacturing, two closely related fields of innovation, offer advantages even over other high technology industries in the United States because they create trade surpluses.[76] For example, cross-border licensing of software generated 36% of all receipts from U.S. trade in intellectual property in 2005, and the resulting surplus of licensing receipts over payments far exceeded the trade surplus in software itself.[77] The value of cross-border software licensing grew dramatically over the last decade, a trend that continues, according to the latest available data.[78] By these measures, international trade in software licenses is of great and growing value to the U.S. economy.

Patents create the intellectual property rights responsible for cross-border software licensing revenue. Software patents are thus valuable in international trade, notwithstanding the possibility that the costs of litigating software patents may reduce their value to software developers.[79] In addition to

---

[75] 1 NAT'L SCI. BD., SCIENCE AND ENGINEERING INDICATORS 2008, at 6-12 to -17, *available at* http://www.nsf.gov/statistics/seind08/pdf/volume1.pdf.

[76] *See* 2 NAT'L SCI. BD., SCIENCE AND ENGINEERING INDICATORS 2008, A6-89 tbl.6-20, http://www.nsf.gov/statistics/seind08/pdf/volume2.pdf (revealing that software and electronics were among only six of eleven high technology industries for which U.S. exports exceeded imports in 2005 and 2006, but that the United States experienced a trade deficit overall in the high technology industries); James E. Bessen & and Robert M. Hunt, *An Empirical Look at Software Patents*, 16 J. ECON. & MGMT. STRATEGY 157, 172 (2007) (demonstrating the importance of software technology to the electronics industry by reporting that 28% of all software patents come from electronics manufacturing firms, compared to only 5% from software publishers during the period 1994-1997).

[77] NAT'L SCI. BD., *supra* note 75, at 6-31 to -32 fig.6-25 (reporting that receipts from cross-border software licensing transactions between unaffiliated companies, which better reflects the negotiated value of the licenses than transactions between parents and subsidiaries, was second only to licensing of industrial processes); NAT'L SCI BD., *supra* note 76, at A6-89 to -94 tbls.6-20 & -23 (reporting that cross-border software licensing receipts of $5.5 billion created a trade surplus of $4.8 billion, compared to only $1 billion surplus exports of software in 2005).

[78] NAT'L SCI. BD., *supra* note 76, at A6-94 tbl.6-23 (revealing a near doubling of both receipts and trade balance from cross-border software licensing over the period 1998 to 2005).

[79] BESSEN & MEURER, *supra* note 46, at 142-46, 143 tbl.6.3, 187-214 (arguing that the costs of litigating software patents exceed their value to the patent owners because abstract

compensating patentees for their research costs through licensing fees and monopoly profits, patents can also offer important strategic benefits if they can be used to protect the patentee's share of the global market against rivals both foreign and domestic.[80] Companies can reap a strategic advantage even if they do not use the patented process or sell the patented device, simply by enforcing the patent to exclude competitors from the market.[81] Software patents are especially susceptible to such strategic use.[82] Thus software patents not only enable substantial licensing revenue from international trade in software, but also provide strategic advantages valuable in both international and domestic markets.

B. *Congressional Protection of Patents in Cross-Border Trade*

Congress has acknowledged the importance of cross-border intellectual property transactions and taken action to protect patentees' revenues derived from the export of patented goods. In response to the Supreme Court's decision in *Deepsouth Packing Co. v. Laitram Corp.*,[83] Congress expressly tightened the patent law to protect technology exporters.[84] *Deepsouth* held that an American competitor did not infringe the patent on a shrimp deveining machine when the competitor manufactured the 1.75 ton machine in pieces, then shipped it overseas to customers who assembled it into the patented whole in less than an hour.[85] The dissenting Justices feared that the decision "would allow an infringer to set up shop next door to a patent-protected inventor

---

claims in software patents provide inadequate notice to potential infringers, resulting in needless infringement and litigation).

[80] Bessen & Hunt, *supra* note 76, at 162 (explaining that a patentee can obtain strategic benefits through "advantages in negotiations, cross-licensing, blocking competitors, and preventing suits" by amassing a defensive portfolio of patents).

[81] *Id.* ("Firms may acquire large numbers of patents so that even if they have an unsuccessful product, they can hold up rivals, threatening litigation.").

[82] *Id.* at 178-79, 184 (concluding that the growth in software patents in the 1990s was largely due to strategic behavior by electronics manufacturers and that software patents actually allowed those manufacturers to decrease their investment in research and development, rather than incentivizing more investment).

[83] 406 U.S. 518, 525-26 (1972).

[84] Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 457 (2007) ("Section 271(f) was a direct response to a gap in our patent law revealed by this Court's Deepsouth decision."); Alan M. Fisch & Brent H. Allen, *The Application of Domestic Patent Law to Exported Software: 35 U.S.C. § 271(f)*, 25 U. PA. J. INT'L ECON. L. 557, 565 (2004) ("Congress specifically intended [35 U.S.C.] § 271(f) as a response to the Supreme Court's decision in *Deepsouth*.").

[85] *Deepsouth*, 406 U.S. at 519-24, 528-32 (holding that a "patent protects only against the operable assembly of the whole and not the manufacture of its parts" and therefore to find infringement by overseas assembly would impermissibly give "extraterritorial effect" to U.S. patent laws).

whose product enjoys a substantial foreign market and deprive him of this valuable business."[86]

Congress responded to this fear by enacting the Patent Law Amendments Act of 1984,[87] which overruled the *Deepsouth* holding by adding a paragraph.[88] The new paragraph, § 271(f)(1), states that:

> Whoever without authority supplies or causes to be supplied in or from the United States all or a substantial portion of the components of a patented invention, where such components are uncombined in whole or in part, in such manner as to actively induce the combination of such components outside of the United States in a manner that would infringe the patent if such combination occurred within the United States, shall be liable as an infringer.[89]

The new subsection took the term "actively induce" from the existing statute that prohibited inducing infringement by "actively and knowingly aiding and abetting another's direct infringement."[90] In the terms of the new statute, Deepsouth Packing Company supplied from the United States all of the components of a patented invention and actively induced their combination outside the United States by overseas customers.[91] Deepsouth would have been liable for contributory infringement under existing law had its customers assembled the components in the United States.[92] Therefore, under § 271(f)(1) Deepsouth would have been equally liable for its customers assembling the patented device overseas. Section 271(f) thus overcame the Supreme Court's strictly territorial interpretation of contributory infringement and protected the economic interests of patentees in cross-border transactions involving their intellectual property.

The *Deepsouth* decision and the congressional response it provoked illustrate the tension between the judicial presumption that U.S. patent law applies only within the territorial limits of the United States and the inequity that patentees sense when U.S. patent law makes it easy for their American

---

[86] *Id.* at 534 (Blackmun, J., dissenting) (internal quotation marks omitted) (citing Laitram Corp. v. Deepsouth Packing Co., 443 F.2d 936, 939 (5th Cir. 1971)).

[87] Pub. L. No. 98-622, 98 Stat. 3383, 3383 (codified at 35 U.S.C. § 271(f) (2006)).

[88] Fisch & Allen, *supra* note 84 at 566 (explaining that Congress enacted § 271(f) as a result of the "inability of the judiciary to prevent patent infringers from" exploiting "limits on extraterritorial application of U.S. patent law").

[89] 35 U.S.C. § 271(f)(1).

[90] *Id.* § 271(b) ("Whoever actively induces infringement of a patent shall be liable as an infringer."); 130 Cong. Rec. 28,069-73 (daily ed. Oct. 1, 1984), *as reprinted in* 1984 U.S.C.C.A.N. 5827, 5828 (providing a section-by-section analysis of patent law amendments of 1984); *see also* C.R. Bard, Inc. v. Advanced Cardiovascular Sys., Inc., 911 F.2d 670, 675 (Fed. Cir. 1990) (applying the § 271(b) active inducement standard to a medical device accused of contributorily infringing).

[91] *See supra* note 85 and accompanying text.

[92] Deepsouth Packing Co. v. Laitram Corp., 406 U.S. 518, 526-27 (1972).

competitors to sell their inventions abroad in a global market of great and growing importance.[93]   Congress went a step further than simply overruling *Deepsouth* when it added paragraph (2) of § 271(f) as part of the 1984 Act:

> Whoever without authority supplies or causes to be supplied in or from the United States any component of a patented invention that is especially made or especially adapted for use in the invention and not a staple article or commodity of commerce suitable for substantial noninfringing use, where such component is uncombined in whole or in part, knowing that such component is so made or adapted and intending that such component will be combined outside of the United States in a manner that would infringe the patent if such combination occurred within the United States, shall be liable as an infringer.[94]

Under paragraph (2), even supplying a single component of a patented invention for subsequent combination outside the United States will render the supplier liable as an infringer unless the component is redeemed by substantial noninfringing uses.[95]   Thus, Congress gave the patent statutes limited extraterritorial effect to ensnare those who would otherwise skirt liability by completing the act of patent infringement overseas.

C. *Extraterritoriality Concerns in Applying § 271(f) to Software Patents*

One might expect that the extraterritorial effect of § 271(f) would apply equally to protect not only patented shrimp deveining machines, but also patented software, a field of invention where cross-border sales and licensing of technology are especially important. However, in *Microsoft Corp. v. AT&T Corp.*,[96] which involved a software patent and was governed by § 271(f), strict territoriality trumped the interests of a software patentee.[97] The Court regarded § 271(f) as "an exception to the general rule that our patent law does not apply

---

[93] Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 454-55 (2007) ("The presumption that United States law governs domestically but does not rule the world applies with particular force in patent law."); *Deepsouth*, 406 U.S. at 531 ("Our patent system makes no claim to extraterritorial effect . . . ."); Brown v. Duchesne, 60 U.S. 183, 195 (1856) ("[The patent laws] do not, and were not intended to, operate beyond the limits of the United States; and as the patentee's right of property and exclusive use is derived from them, they cannot extend beyond the limits to which the law itself is confined.").

[94] Pub. L. No. 98-622, 98 Stat. 3383, 3383 (codified at 35 U.S.C. § 271(f)).

[95] *See Microsoft v. AT&T*, 550 U.S. at 454 n.16 (concluding that distinctions between the two paragraphs did not affect the outcome of that case).

[96] *Id.* at 453-54 (holding that a software "master disk" did not qualify as a "component" within the meaning of § 271(f)).

[97] *Id.* at 456-57 (acknowledging that a strictly territorial interpretation of § 271(f) creates a loophole for software makers); *see supra* Part II.B.

extraterritorially" and consequently interpreted the *Deepsouth* statutory exception narrowly.[98]

In *Microsoft v. AT&T*, plaintiff AT&T patented a device that combines a computer and a piece of software to digitally encode recorded speech.[99] Defendant Microsoft incorporated the AT&T software in its operating system and sent it abroad intending that it be combined with computers that were then sold overseas.[100] On their face, these facts sound analogous to *Deepsouth*, and seem to embody precisely the conduct that Congress intended to outlaw when it overruled *Deepsouth* by enacting § 271(f).[101]

Before the case reached the Supreme Court, the Court of Appeals for the Federal Circuit concluded that Microsoft had indeed violated § 271(f).[102] The two pivotal questions in the case were (1) whether the Microsoft software was a component of the foreign-made computers for purposes of § 271(f) and, if so, (2) whether that component was supplied from the United States.[103]   The Federal Circuit answered yes to both questions and concluded that Microsoft had therefore infringed the AT&T patent under § 271(f), notwithstanding the fact that Microsoft had sent the software to the foreign computer manufacturers "on so-called 'golden master' disks or via electronic transmissions," which had to be replicated to create copies installable on individual computers.[104]

The Federal Circuit had recently prepared the way for this result in a case involving similar facts, *Eolas Technologies Inc. v. Microsoft Corp*.[105] In this case, the court construed § 271(f) not to require that a component be a tangible, physical or structural piece of the patented invention, thereby making software eligible as a component.[106] Furthermore, software made in the United States and sent abroad on a golden master disk was a component of the computers on which the software ultimately was installed – regardless of the intermediate

---

[98] *Microsoft v. AT&T*, 550 U.S. at 441-42; *see* 35 U.S.C. § 271(a) (illustrating the general rule: "whoever without authority makes, uses, offers to sell, or sells any patented invention, *within the United States* . . . infringes the patent" (emphasis added)).

[99] *Microsoft v. AT&T*, 550 U.S. at 441-42.

[100] *Id.* (explaining that the "patent is infringed only when a computer is loaded with Windows and is thereby rendered capable of performing as the patented speech processor").

[101] Although AT&T never specified which paragraph of § 271(f) prohibited Microsoft's conduct, paragraph (2) is more applicable since Microsoft did not necessarily supply all components of the invention. *See id.* at 463 (Stevens, J., dissenting) (concluding that Microsoft had infringed the AT&T patent under § 271(f)(2)); *supra* note 95.

[102] Timothy R. Holbrook, *Extraterritoriality in U.S. Patent Law*, 49 WM. & MARY L. REV. 2119, 2132-36, 2146-48 (2008) ("Before the Supreme Court reined in the scope of § 271(f) to some extent in *Microsoft Corp. v. AT&T Corp.*, the Federal Circuit consistently gave that provision a rather broad scope.").

[103] AT&T Corp. v. Microsoft Corp., 414 F.3d 1366, 1369 (Fed. Cir. 2005).

[104] *Id.* at 1367-69.

[105] 399 F.3d 1325, 1339 (Fed. Cir. 2005).

[106] *Id.* ("[E]very component of every form of invention [including software] deserves the protection of section 271(f).").

step of replicating the master disk.[107]   The court found no basis to treat intangible software differently from a physical component, especially since a functioning computer "transforms the code on the golden disk into a machine component in operation."[108]

Once the Federal Circuit, relying on *Eolas*, concluded that intangible software was the component at issue, the dispositive fact was that the intangible software traveled from the United States to the ultimate computers, a fact not altered by a replication step subsumed in the process of supplying the software.[109] Thus, Microsoft's conduct infringed AT&T's patent. Dissenting Judge Rader maintained the contrary view that replication of the master disk amounted to a distinct manufacturing step creating CD-ROM components, not supplied by Microsoft, which the foreign manufacturer ultimately combined with individual computers to make the patented speech-encoder.[110] On this reasoning, the software component had not been supplied from the United States, so liability did not result.[111]

Nevertheless, the Federal Circuit majority did not hesitate to find that § 271(f) had sufficient extraterritorial reach to make Microsoft liable for conduct that included a step occurring entirely on foreign soil.[112] The majority argued that this result was consistent with "the statute's purpose of prohibiting the circumvention of infringement through exportation."[113]

The Supreme Court reversed every step of the Federal Circuit's seemingly cogent reasoning, holding that intangible software is not a component of a tangible invention under § 271(f), and therefore the components of AT&T's invention were manufactured abroad instead of supplied from the United

---

[107] *Id.* at 1339-40 ("[M]uch more than a prototype, mold, or detailed set of instructions . . . the software code on the golden master disk is not only a component, it is probably the key part of this patented invention.").

[108] *Id.* at 1339.

[109] *AT&T v. Microsoft*, 414 F.3d at 1370 ("Given the nature of the technology, the 'supplying' of software commonly involves generating a copy.").

[110] *Id.* at 1373 (Rader, J., dissenting) ("[C]opying and supplying are separate acts with different consequences – particularly when the 'supplying' occurs in the United States and the copying occurs in Düsseldorf or Tokyo.").

[111] *Id.* (finding that making installable copies from the Microsoft golden master disk violated § 271 but that exporting the golden master disk did not).

[112] *Id.* at 1371 (majority opinion) (holding that Microsoft, in supplying software "specifically for the purpose of foreign replication," could not escape liability without subverting the nature of § 271(f)).

[113] *Id.* at 1368, 1371 (affirming the district court ruling, refusing "to hold that Microsoft's supply by exportation of the master versions . . . avoids infringement").

States.[114] Moreover, the Court's presumption against extraterritoriality sealed the conclusion that Microsoft did not infringe.[115]

The Court distinguished between software in the abstract and tangible copies of software.[116] It characterized Microsoft's software, "abstracted from a tangible copy," as information similar to a blueprint, which might contain instructions for a patented device but "is not itself a combinable component of that device."[117] "Until it is expressed as a computer-readable 'copy,' *e.g.*, on a CD-ROM, Windows software – indeed any software detached from an activating medium – remains uncombinable."[118] Thus, AT&T could not argue that its software must have been combined with the overseas computers simply because the software was found on those computers. In the Court's reasoning, software as an abstract set of instructions differs from a specialized crankshaft made in the United States, which of necessity can only be found in an engine overseas if it was somehow combined with that engine. The software itself cannot travel as a disembodied set of instructions – to be a component, it must reside on a physical medium. The Court left open the possibility that software in the abstract could be a component of an invention that was itself an intangible process, not reaching that question because "AT & T's speech-processing computer . . . is a tangible thing."[119]

On the question of whether Microsoft had *supplied* a software component from the United States, the Supreme Court adopted Judge Rader's reasoning that copying the master disk constituted a manufacturing step distinct from the supplying of the software.[120] If supplying is distinct from the subsequent step of copying, and only the tangible copies created in the copying step are components, then the foreign manufacturers combined computers with software components manufactured abroad – not with any component supplied from the United States by Microsoft.[121]   In other words, Microsoft's

---

[114] Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 449-56 (2007) (discussing that abstract software code "detached from an activating medium" falls outside of § 271(f)'s reach).

[115] *Id.* ("Any doubt that Microsoft's conduct falls outside § 271(f)'s compass would be resolved by the presumption against extraterritoriality . . . .").

[116] *See supra* note 5 and accompanying text (introducing the distinction between tangible software and software in the abstract).

[117] *Microsoft v. AT&T*, 550 U.S. at 449-50.

[118] *Id.* (holding that software in the abstract "is an idea without physical embodiment, and as such, it does not match § 271(f)'s categorization: 'components' amenable to 'combination'").

[119] *Id.* at 452 n.13 (noting that combinable components of an intangible process "might be intangible as well," but reaching no conclusion on that issue, and distinguishing the facts of this case).

[120] *Id.* at 452-54 (arguing that the ease with which software can be copied does not justify subsuming the act of copying into the act of supplying); *see also supra* note 110 (discussing Judge Rader's view).

[121] *Microsoft v. AT&T*, 550 U.S. at 452-54.

contribution to the foreign-built, speech-processing computers was not a component, but a piece of intangible, uncombinable software borne overseas by the master disk. While it may have "facilitate[d] making a component of a patented invention outside the United States," Microsoft supplied no component from the United States, and therefore was not liable for infringement under § 271(f).[122]

Whereas the Federal Circuit supported its finding of infringement by construing the extraterritorial reach of § 271(f) as sufficient to follow intangible software sent abroad, the Supreme Court supported its finding of noninfringement by construing the same language narrowly, to avoid ensnaring acts of software-copying abroad.[123] The Supreme Court acknowledged that its interpretation left a loophole for "an easily arranged circumvention" of § 271(f), but deferred to Congress to close that loophole.[124]

III. DO PATENTS PROTECT SOFTWARE? – WOULD COPYRIGHTS DO BETTER?

A. *Effects of the* Microsoft v. AT&T *Loophole*

Juxtaposing the widely divergent Federal Circuit and Supreme Court opinions in *Microsoft v. AT&T* demonstrates that a patent on a device incorporating software provides varying protection to the patent holder according to the court's choice among a range of reasonable characterizations of software. The choice of characterization directly affected the exceptionally valuable right of software and electronics inventors to exclude domestic competitors from profiting by foreign sales of their inventions.[125]

The Supreme Court's holding that software transmitted by master disk or electronic means does not constitute a component of a patented device is a setback for patentees seeking protection in cross-border trade.[126] While the majority does not reach the question, three Justices in concurrence point out that the majority's reasoning demands that if a master disk is not a component, then neither is a CD-ROM copy of the master disk bearing the same intangible software and sent abroad to install the software directly on an individual computer.[127] Either way, no physical component supplied from the United

---

[122] *Id.* at 458.

[123] *Id.* at 456-58 (acknowledging that the narrow reading of § 271(f) "cover[s] only those copies of software actually dispatched from the United States").

[124] *Id.* at 456.

[125] *See supra* Part II.A (discussing the value of this right, especially in light of such considerations as substantial licensing revenues).

[126] *See supra* text accompanying note 104 (recounting that Microsoft employed both golden master disks and electronic transmission to send its software to foreign manufacturers).

[127] *Compare Microsoft v. AT&T*, 550 U.S. at 453 n.14 (declining to reach the question of whether liability under § 271(f) would result if Microsoft shipped disks from the United States that foreign manufacturers used directly to install on their computers), *with id.* at 461-

States is ever combined in the ultimate device, since the CD-ROM is removed after installation.[128] The concurring Justices allow that if the CD-ROM needed to remain in the computer to run the software, then it might be a component.[129]

The majority holding and its logical elaboration by the concurrence give rise to the following results.[130] Software in the abstract cannot be a component of a patented device.[131] Software transmitted electronically cannot be a component of a patented device.[132] Software on a CD-ROM cannot be a component of a patented device.[133] However, software installed on a hard drive is a component of the patented device, since it needs to remain in the computer to run.[134] Thus, software in any transportable form is an uncombinable non-component, and yet ultimately a manufacturer combines it as a component in the complete, patented device.

This result limits patentees' protection against domestic competitors who arrange to manufacture abroad a patented device incorporating software, but leaves intact the protection against analogous infringement occurring wholly within the United States. A domestic competitor can infringe a patent directly by making, using, or selling the patented device, or indirectly by inducing or contributing to infringement.[135] A competitor who domestically manufactures an entire patented device including software is liable for direct infringement under § 271(a) regardless of the manufacturing steps taken to combine the software in the device.[136] Likewise, a domestic competitor who knowingly aids and abets a domestic third party in manufacturing the device by licensing the necessary software is liable for inducing infringement under § 271(b),

---

62 (Alito, J., concurring) (finding no liability under § 271(f) even without the intermediate step of replicating the master disk).

[128] *Id.* at 461-62 ("Because the physical incarnation of code on the Windows CD-ROM supplied from the United States is not a 'component' of an infringing device under § 271(f), it logically follows that a copy of such a CD-ROM also is not a component.").

[129] *Id.* ("To be sure, if these computers could not run Windows without inserting and keeping a CD-ROM in the appropriate drive, then the CD-ROMs might be components of the computer. But that is not the case here.")

[130] *Id.* at 450-52 (majority opinion) (holding that software in the abstract is not a combinable component); *id.* at 461 (Alito, J., concurring) (elaborating that a copy of a CD-ROM that is not a component is also not a component).

[131] *Id.* at 451-52 (majority opinion).

[132] *See id.* at 445 (describing the process by which Microsoft electronically transmits the masters).

[133] *Id.* at 461-62 (Alito, J., concurring).

[134] *Id.*

[135] *See* 35 U.S.C. § 271 (2006) (setting forth what constitutes patent infringement).

[136] *Microsoft v. AT&T*, 550 U.S. at 446 & n.5 (relating Microsoft's stipulation that by installing the software on its own computers it had directly infringed AT&T's patent under § 271(a)).

regardless of how the competitor transmits the software to the manufacturer.[137] Finally, a competitor who sells software in the United States serving no purpose other than to complete the patented device would not escape liability for contributory infringement under § 271(c) even if courts construed the term "component of a patented machine" in § 271(c) the same way that *Microsoft v. AT&T* construed the term "component of a patented invention" in § 271(f).[138] Even if the software is sold domestically in a form that would not qualify as a component of the patented invention if supplied overseas, the question of supply never arises under § 271(c), and liability results simply because the ultimate patented device incorporates the software.[139]

Thus, the conclusion that intangible software cannot be a component of a patented device, even if applied outside § 271(f), only affects software in cross-border trade. Furthermore, software-based devices get less patent protection than other non-structural components in the context of cross-border trade.[140] Unlike software, other non-structural components like chemical additives may not be assembled overseas into a patented device without

---

[137] *See id.* at 446 & n.6 ("Microsoft further acknowledged that by licensing copies of Windows to manufacturers of computers sold in the United States, it induced infringement of AT&T's patent."); *supra* note 90 and accompanying text (introducing this theory of infringement for aiding and abetting).

[138] *Compare* 35 U.S.C. § 271(c) (prohibiting contributory infringement occurring wholly within the United States), *with id.* § 271(f)(2) (prohibiting analogous acts of contributory infringement involving supply of a component from the United States and combination with other components of an invention overseas).

[139] The question of when software becomes a combinable component owes its significance to the follow-up question of whether the component was supplied from the United States, which occurs only in the context of cross-border trade. *See supra* text accompanying note 103 (outlining these two inquiries in the context of *Microsoft v. AT&T*); *see also* Ricoh Co. v. Quanta Computer Inc., 550 F.3d 1325, 1336-40 (Fed. Cir. 2008) (vacating summary judgment because there was no contributory infringement by optical disc drive manufacturer Quanta on grounds that "Quanta's drives use separate hardware and embedded software modules to perform the patented processes and that those components had no noninfringing use," regardless of how those modules were supplied). Microsoft wasted no time in attempting to use the *Microsoft v. AT&T* holding to overturn a jury verdict of contributory infringement under § 271(c) in a separate case involving domestic sales of a patented audio compression encoder. *See* Lucent Techs. Inc. v. Gateway, Inc., 509 F. Supp. 2d 912, 929-30 (S.D. Cal. 2007) (denying Microsoft's motion for a new trial on grounds that while "[o]ne of the key concerns regarding § 271(f) is the effect of U.S. patent law on extraterritorial activities[,] . . . [t]his concern does not infect § 271(c)," and thus there is neither rationale nor "precedent for limiting the scope of § 271(c) to the limits placed on § 271(f)"), *aff'd on other grounds*, 543 F.3d 710, 724 (Fed. Cir. 2008).

[140] *See* W.R. Grace & Co.-Conn. v. Intercat, Inc., 60 F. Supp. 2d 316, 320 (D. Del. 1999) (concluding that § 271(f) is not limited to "components of machines and other structural combinations," thereby finding liability for combining chemical compositions overseas).

violating § 271(f).[141] Some would go so far as to say that *Microsoft v. AT&T* specifically targeted software inventors and deprived them of any meaningful protection under § 271(f).[142]

It is inaccurate to say that the decision eliminated all protection since inventors might still be able to protect software as part of a patented method rather than part of a patented device.[143] Even as the Court in *Microsoft v. AT&T* held that intangible software could not be a component of a patented device, it specifically left open the possibility that software might qualify as a component of an intangible method patent.[144] This dictum undermines previous efforts by lower courts to foreclose alternative protection for exported software under method patents instead of device patents.[145] According to those decisions, methods fell outside § 271(f) because they lacked component parts.[146] *Microsoft v. AT&T* expressly declined to endorse this reasoning and thereby kept open the door to protecting intangible software from combination abroad into a patented method.

B. *Combining* Microsoft v. AT&T *with* In re Bilski

Against this backdrop, the Federal Circuit's decision in *In re Bilski* pushes to close the door on software patents again.[147] It held that a method patent comprising an algorithm unconnected to a specific physical transformation is not patentable.[148] Software is an algorithm, and a method comprising software running on a computer is easily seen as an algorithm unconnected to a specific physical transformation, and therefore unpatentable.[149] Thus, while *In re Bilski*

---

[141] *Id.* at 319-21 (holding that assembling chemical additives abroad would violate § 271(f)).

[142] *See* Erika Danielle Norman, *Weak Overseas Protection for American Software Patents: The Need for a Congressional Response to* Microsoft Corp. v. AT&T Corp., 8 CHI.-KENT J. INTELL. PROP. 111, 132-33 (2008) ("The Court's holding allows Microsoft and other U.S.-based software companies to avoid liability under § 271(f) by supplying software abroad for copying and installation.").

[143] 35 U.S.C. § 101 (authorizing patents for machines, manufactures, compositions of matter, and processes or methods).

[144] *See supra* note 119 and accompanying text (explaining that combinable components of an intangible process might also be intangible).

[145] *See, e.g.*, Enpat, Inc. v. Microsoft Corp., 6 F. Supp. 2d 537, 539 (E.D. Va. 1998) (granting partial summary judgment of noninfringement when software was sold abroad to practice a computer-based method patent because the patent had "no 'components' for purposes of § 271(f)").

[146] *See* Fisch & Allen, *supra* note 84, at 570-72.

[147] *See supra* notes 68-72 and accompanying text (explaining that *In re Bilski* left the patentability of software as method patents an open question but that the machine-or-transformation test makes patentability unlikely).

[148] *See supra* notes 63-67 and accompanying text (discussing the machine-or-transformation test and how it limits patentability).

[149] *See supra* Part I.A.

does not reach the question of whether software as part of a computer is patentable subject matter,[150] it holds that software as part of a method that does not perform a physical transformation is not. Even if the Supreme Court in *Microsoft v. AT&T* did not rule out a software component as part of a method patent, the Federal Circuit rendered many software-based methods unpatentable under *In re Bilski*.

Where does this leave a company like AT&T? Consider an American inventor, similar to the one in *Microsoft v. AT&T*, who develops a piece of software that, when used on a personal computer, translates audio files of German speech into files of German text. Assuming that the invention satisfies the requirements of a U.S. patent, what protection would a patent afford against a American competitor who is selling the software in Europe, the only market for such software? The inventor has three possible paths to patent protection: 1) patent the translation algorithm itself, 2) patent the translation device comprising the software and a personal computer, and 3) patent the translation method that uses the software on a personal computer. The inventor is free to pursue any or all of these paths, by applying for one or more patents, but this would be to no avail under *In re Bilski* and *Microsoft v. AT&T*.

First, patenting the translation algorithm itself would make the competitor liable under § 271(a) for making or using the software in the United States.[151] The inventor would then have a remedy if the competitor loaded the software on computers in the United States and shipped them to Germany or sent the software by itself, whether via email, compact disk, or golden master disk. However, the Federal Circuit would likely not enforce a patent on the translation software in the abstract, notwithstanding the absence of a per se rule against software patents in *In re Bilski*.[152] Software in the abstract lacks the necessary connection to a machine or transformation and is essentially similar to Bilski's method of calculating commodity transactions, rendering it likely unpatentable.[153]

Next, patenting the translation device comprising the software and a personal computer creates potential liability under not only § 271(a) but also § 271(f) because the device is a combination. Under § 271(a), the inventor would have a remedy only if the competitor loaded the software on computers in the United States and shipped them to Germany.[154] However, the competitor could easily circumvent that liability by shipping only the software

---

[150] *See supra* notes 68-72 and accompanying text (remarking that the court in *In re Bilski* does not directly address whether software is patentable as part of a machine when it adopted the machine-or-transformation test).

[151] 35 U.S.C. § 271(a) (2006) ("[W]hoever without authority makes, uses, offers to sell, or sells any patented invention, within the United States . . . infringes the patent").

[152] *See supra* notes 68-72 and accompanying text.

[153] *See supra* notes 63-67 and accompanying text (discussing the application of the machine-or-transformation test to Bilski's method patent).

[154] *See* 35 U.S.C. § 271(a).

to Germany and assembling the patented combination there. Section 271(f) would make the competitor liable as an infringer if the software is a component of the patented device.[155] Unfortunately for the inventor, *Microsoft v. AT&T* makes it clear that software supplied from the United States by any medium is not a component of a patented device for purposes of § 271(f), and thus the competitor escapes liability.[156]

Finally, patenting the translation method consisting of using the software on a personal computer would also create potential liability under both § 271(a) and § 271(f). Under § 271(a), the competitor would be liable if he practiced all steps of the patented method in the United States by combining the software with a personal computer in the United States.[157] The competitor could again seek to circumvent § 271(a) liability by shipping the software to Germany. In either case, it is unlikely that the Federal Circuit would enforce the patent in light of *In re Bilski.* The court would likely see a method to translate computer representations of spoken words into computer representations of written words as fundamentally equivalent to Bilski's method of calculating commodity transactions, and therefore unpatentable.[158]

The transformation branch of the *Bilski* test excludes a significant class of software from patentability.[159] That class includes not only translation software, but word processors as well, since words, after all, are not physical objects. Likewise, economic abstractions such as risks and commodity prices are not physical objects.[160] Methods for manipulating and displaying general, unspecified data, as in spreadsheet software, also fail the transformation test because the data is not tied to a specific physical object.[161] Moreover, even software that passes the transformation test would not be patentable in the abstract under *In re Bilski*, nor would it be protected against overseas combination in a patented device under *Microsoft v. AT&T.* Thus, the combined result of the holdings in *Microsoft v. AT&T* and *In re Bilski* is that a significant class of software inventors will have no remedy at patent law against an American competitor who sells their inventions overseas in what may be the only market for them.

---

[155] *See supra* text accompanying notes 89-95.

[156] *See supra* Part III.A.

[157] NTP, Inc. v. Research in Motion, Ltd., 418 F.3d 1282, 1318 (Fed. Cir. 2005) ("[A] process cannot be used 'within' the United States as required by section 271(a) unless each of the steps is performed within this country.").

[158] *See supra* notes 68-72 and accompanying text (explaining that after *In re Bilski* it is unlikely that such software could be protected as a method patent).

[159] *See supra* note 70 and accompanying text.

[160] *In re* Bilski, 545 F.3d 943, 963-64 (Fed. Cir. 2008) ("Purported transformations or manipulations simply of public or private legal obligations or relationships, business risks, or other such abstractions cannot meet the test because they are not physical objects or substances, and they are not representative of physical objects or substances.").

[161] *Id.* at 962 (explaining that a claim that does not specify the nature of the data or what the data represented was unpatentable).

C.	*Alternative Remedies for Software Patentees*

In declining to find liability for circumventing U.S. patents by exporting software components, American courts have suggested that patentees should instead seek their remedy at foreign patent law.[162]	The result is that the patentee of a device incorporating software must obtain patents and litigate them separately in every nation where the patentee fears infringement. While this may be good advice for patents on some types of inventions, a remedy at foreign law may be less effective when the inventions involve software.

The European Patent Convention, covering thirty-five European member states, specifically excludes "programs for computers" from the definition of patentable inventions.[163]	The European Patent Office Technical Board of Appeal nevertheless has decided to grant patents for inventions in which a computer program is necessary to achieve an effect "by the internal functioning of a computer itself under the influence of said program."[164]	In principle, it should be easier to obtain a software patent under this rule than under the *In re Bilski* machine-or-transformation test. In the language of the *Bilski* test, the European Patent Office would recognize software functioning on a general purpose computer as having a sufficient connection to a machine to render the computer plus software combination patentable per se.[165]	The Federal Circuit refused to go so far in *In re Bilski*.[166]	However, programs for computers nevertheless remain unpatentable subject matter according to the

---

[162] *See, e.g.*, Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 456 (2007) ("If AT&T desires to prevent copying in foreign countries, its remedy today lies in obtaining and enforcing foreign patents."); Deepsouth Packing Co. v. Laitram Corp., 406 U.S. 518, 531 (1971) ("To the degree that the inventor needs protection in markets other than those of this country, the wording of 35 U.S.C. §§ 154 and 271 reveals a congressional intent to have him seek it abroad through patents secured in countries where his goods are being used."); AT&T Corp. v. Microsoft Corp., 414 F.3d 1366, 1375 (Fed. Cir. 2005) (Rader, J., dissenting) ("AT & T can protect its foreign markets from foreign competitors by obtaining and enforcing foreign patents.").

[163] Convention on the Grant of European Patents, art. 52(2), Oct. 5, 1973, 1065 U.N.T.S. 255 (regarding "schemes, rules and methods for performing mental acts, playing games or doing business, and programs for computers" as unpatentable); European Patent Office, Member States of the European Patent Organisation, http://www.epo.org/about-us/epo/member-states.html (last visited May 15, 2009).

[164] Case T-1173/97, Computer Program Prod./IBM 15-16 (European Patent Office Technical Board of Appeal 1998), *available at* http://legal.european-patent-office.org/dg3/pdf/t971173ex1.pdf.

[165] *See supra* notes 63-67 and accompanying text (describing the machine-or-transformation test).

[166] *Supra* note 68 and accompanying text (remarking that the court in *In re Bilski* withheld judgment on the issue).

letter of the European Patent Convention, and European authorities continue to debate whether software should be patentable subject matter.[167]

Indigenous patent protection is even less likely to be available to U.S. software inventors in less developed nations. For instance, while India is thought to provide stronger intellectual property protection than other important destinations for outsourcing, such protection is significantly weaker than in the United States or Europe.[168] In particular, India maintains a per se rule against the patentability of computer programs.[169] Obtaining foreign patents is thus a dubious strategy for an American software inventor.

It is quite possible that the only acts of infringement taking place in the foreign country are the direct acts of consumers loading the software onto their personal computers. This problem would arise if Microsoft were to sell CD-ROMs of software containing AT&T's speech encoder overseas where consumers would install it on their computers, creating AT&T's patented invention.[170] Even if consumers were liable under foreign patent law, seeking a remedy against those direct infringers may not be practical if they are too numerous.[171] AT&T would likely prefer to pursue a claim of contributory infringement against the distributor Microsoft.[172] Whether AT&T could succeed in a foreign venue would depend not only on the merits of its contributory infringement claim in light of foreign law but also on whether the foreign court would assert jurisdiction over Microsoft. Thus, when the U.S. inventor of a software-based invention finds an American competitor making the invention overseas, the inventor likely will not find an effective remedy

---

[167] Violeta I. Balan et al., *International Legal Developments in Review: 2007 Regional and Comparative Law – Europe*, 42 INT'L LAW. 975, 993 (2008) ("The test for the patentability of software and business methods continues to be the subject of debate in Europe.").

[168] Sonia Baldia, *Navigating Cross Border Legal Risks in Intellectual Property Licensing and Technology Transfer to India*, 1720 PLI/CORP 191, 207 (2009) ("While India is perceived as providing better IP protection compared to a majority of the other jurisdictions, such as China, Russia and Mexico, India's IP laws and enforcement mechanisms are nevertheless weak compared to western IP laws and practices . . . .").

[169] *Id.* at 214 ("Computer programs and business methods continue to be *per se* not patentable in India . . . .").

[170] *See* Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 445-46 (2007) (articulating the similar, but slightly different facts of the actual case, in which Microsoft sent only a single disk or electronic transmission abroad containing the software that completed the AT&T invention).

[171] *See* Metro-Goldwyn-Mayer Studios Inc. v. Grokster, Ltd., 545 U.S. 913, 929-30 (2005) (finding a strong argument to impose indirect liability "[w]hen a widely shared service or product is used to commit infringement, [because] it may be impossible to enforce rights in the protected work effectively against all direct infringers").

[172] *See id.* (explaining that indirect liability against the distributor might be preferable in such a scenario).

under the local patent laws of the country where the infringement is taking place.

D. *Recommendations*

The holdings of *Microsoft v. AT&T* and *In re Bilski*, taken together, effectively eliminate protection under domestic patent law for an American software inventor against domestic competitors who export the software to overseas markets.[173] Remedy under foreign patent law is also problematic.[174] The lack of protection of software patent rights across national borders, in an age when transnational sales of software products is of huge and growing importance, seriously diminishes an important commercial benefit of software patents.[175]

It may be possible to improve protection for intellectual property rights of U.S. software sold abroad either by tightening the patent law or by looking outside the patent law. Several commentators have called upon Congress to legislate a solution within the patent law by abrogating the *Microsoft v. AT&T* holding and expressly recognizing software as a component under 35 U.S.C. § 271(f).[176] Alternatively, patent protection for software could be left to atrophy, encouraging U.S. software inventors to seek more effective protection from copyright under both domestic and international law.

1. Improving Patent Protection

The proposal to legislatively expand the definition of a component to include software would no doubt close the loophole in § 271(f) that was opened by *Microsoft v. AT&T*, and thereby sidestep the retrenching effects of *In re Bilski*. However, *Microsoft v. AT&T* joins at least two recent Federal Circuit decisions that purposely avoided giving cross-border effect to U.S. patent law.[177] This tendency toward strict territoriality is supported by

---

[173] *Supra* Part III.B (discussing the combined result of the two cases and where it leaves a company like AT&T).

[174] *Supra* Part III.C.

[175] *See supra* Part II.A (discussing the large economic interests at stake in global manufacturing and licensing).

[176] *E.g.,* Norman, *supra* note 142, at 137-39 ("Congress should amend § 271(f) by adding a new paragraph, (f)(3), to read 'for purposes of this section, the term "component" includes intangible software components even if such components are staple articles or commodities of commerce suitable for substantial non-infringing use.'"); Peter Thomas Luce, Comment, *Hiding Behind Borders in a Borderless World: Extraterritoriality Doctrine and the Inadequacy of U.S. Software Patent Protections in a Networked Economy*, 10 TUL. J. TECH. & INTELL. PROP. 259, 285 (2007).

[177] Microsoft Corp. v. AT&T Corp., 550 U.S. 437, 441-42 (2007) (holding that intangible software is not a patentable component in international trade); Voda v. Cordis Corp., 476 F.3d 887, 890-91 (Fed. Cir. 2007) (declining to assert supplemental jurisdiction over a U.S. citizen's claims arising from his foreign patents in a case about his domestic patent on the same product); NTP, Inc. v. Research in Motion, Ltd., 418 F.3d 1282, 1318

underlying considerations of comity, judicial economy, convenience and fairness, and is therefore unlikely to reverse.[178] Thus, even if Congress were to bolster patent protection for software, the background judicial presumption denying extraterritorial effect to the patent law would tend to curb expanded protection in the field of cross-border trade.

In addition, strengthening software patents would exacerbate some of their negative consequences. The cost of litigating patent disputes exceeds the revenues attributable to patents in several industries, including information and communications technology.[179] Experience with the open source software movement casts doubt on the utilitarian rationale for granting patent protection in the software field.[180] Furthermore, the present approach to software patents fails to promote software security, an increasingly important factor in social utility.[181] Finally, ongoing efforts to harmonize patent law internationally threaten negative consequences in both the United States and the rest of the world if the existing patent regime in the United States is propagated globally.[182] Thus, the background judicial presumption that patents should not be permitted extraterritorial effect, along with policy concerns about expanding the scope of software patent rights, suggests that a better solution is to look beyond patent protection for intellectual property rights in software.

2.   Relying on Copyright Protection

How would the existing regime of domestic copyright protection, coupled with harmonized international copyright law, protect our developer of software that translates German audio files into German text files? Under U.S. copyright law, the translation software in the abstract – the text of the software

---

(Fed. Cir. 2005) (holding that the use of a method is not within the U.S. for infringement purposes unless all the claimed steps are performed in the U.S.). *NTP v. Research in Motion* distinguished method patents from system or device patents, for which it allowed some extraterritorial effect in cases of large, geographically distributed systems. *Id.* at 1317-18 ("Under section 271(a), the concept of 'use' of a patented method or process is fundamentally different from the use of a patented system or device.").

[178] *Voda*, 476 F.3d at 900-04.

[179] BESSEN & MEURER, *supra* note 46, at 15 (reporting that since the early 1990s, the aggregate cost of litigating patents in the U.S. has exceeded the profit associated with those patents except in the chemical and pharmaceutical industries).

[180] Benkler, *supra* note 39, at 444-46 (describing commons-based peer production as an effective creative engine that does not depend on traditional property rights); *see supra* Part I.D (describing how granting property rights in software generates controversy).

[181] Jay P. Kesan & Rajiv C. Shah, *Shaping Code*, 18 HARV. J.L. & TECH. 319, 388-92 (2005) (arguing that intellectual property rights in software should be manipulated to achieve social benefits through "code shaping").

[182] Jerome H. Reichman & Rochelle Cooper Dreyfuss, *Harmonization Without Consensus: Critical Reflections on Drafting a Substantive Patent Law Treaty*, 57 DUKE L.J. 85, 108-09 (2007) (predicting adverse impact on the information technology sector in developed countries).

itself – is protected if it is fixed in any tangible form, whether paper, computer memory chip, computer hard drive, or compact disc.[183] The software is protected against direct literal copying and, if a competitor made minor changes, against substantial non-literal copying as well.[184] By contrast, under U.S. patent law, protection is likely not available for software in the abstract, and the inventor is forced to try to protect it as part of a hardware device.[185] Under the copyright regime, the software itself is protected, whether it is operating on a computer or sitting inert on a disk. And to enforce that protection, the inventor can turn to digital rights management technology such as digital watermarking to help identify copied material and pursue unauthorized copyists.[186] A competitor who reproduces the translation software on compact discs in the United States and then exports them will thus be liable for copyright infringement but not for patent infringement.

Furthermore, despite the "undisputed axiom that United States copyright law has no extraterritorial application,"[187] the inventor can pursue extraterritorial damages if the copyrighted software is exploited overseas. The only two circuits to consider the availability of extraterritorial damages for domestic copyright infringement have allowed such damages.[188] The courts have followed Judge Learned Hand's reasoning in the Second Circuit that the copyright owner "acquired an equitable interest in [infringing works] as soon as they were made, which attached to any profits from their exploitation, whether in the form of money remitted to the United States, or of increase in

---

[183] MAI Sys. Corp. v. Peak Computer, Inc., 991 F.2d 511, 518 (9th Cir. 1993) (holding that "'copying' for purposes of copyright law occurs when a computer program is transferred from a permanent storage device to a computer's RAM," so that copies on both the electronic storage device and the RAM are protectable).

[184] *See supra* Part I.B (chronicling the expansion of software copyright doctrine to cover non-literal copying in the 1980s and 1990s).

[185] *See supra* Part I.E (discussing that software in the abstract is likely not patentable after *In re Bilski*).

[186] Terri Branstetter Cohen, Note, *Anti-Circumvention: Has Technology's Child Turned Against Its Mother?*, 36 VAND. J. TRANSNAT'L L. 961, 973-74 (2003) ("Digital watermarks contain data, such as copyright information, that identifies a work and is incorporated into the work itself; watermarking allows the content owner to track the use of his work . . . .").

[187] Subafilms, Ltd. v. MGM-Pathe Commc'ns Co., 24 F.3d 1088, 1093 (9th Cir. 1994) (quoting 3 DAVID NIMMER & MELVILLE B. NIMMER, NIMMER ON COPYRIGHT § 12.04[A][3][b], at 12-86 (1991)).

[188] *See* Los Angeles News Serv. v. Reuters Television Int'l, Ltd., 149 F.3d 987, 992 n.3 (9th Cir. 1998) ("No other circuits appear to have addressed this issue."), *aff'd*, 340 F.3d 926, 931-32 (9th Cir. 2003) (limiting recovery of extraterritorial damages to the infringer's profits from the overseas exploitation, not the actual damages resulting from the infringement); Update Art, Inc. v. Modiin Pub., Ltd., 843 F.2d 67, 73 (2d Cir. 1988) (holding that in the absence of evidence that "the predicate act of reproducing the poster occurred in Israel[;] . . . [d]amages accruing from the illegal infringement in the Israeli newspapers properly were awarded to [the copyright owner]").

the value of shares of foreign companies held by the defendants."[189] Judge Hand's equitable principle might seem to apply equally well to patent infringement, but it has not been extended to patent cases.

Judge Hand held that the copyright owner was entitled to damages for infringement of his copyrighted play when negatives of the infringing motion picture were sent overseas and copied.[190] Likewise, when a Reuters subsidiary transmitted copyrighted news video to overseas subscribers without authorization, the Ninth Circuit held Reuters liable for the profits it made from the broadcast.[191] The subsidiary made a videotape copy of the work and then made a profit by transmitting it via satellite to subscribers in Europe and Africa.[192] Notwithstanding the settled law that "extraterritorial infringement does not violate American copyright law,"[193] the copyright owner was entitled to damages equal to the profits from the overseas subscriptions, on grounds that the extraterritorial damages stemmed from domestic infringement.[194] Our software inventor could likewise argue that he is entitled to any profits that a competitor receives from transmitting the translation software to customers abroad. By analogy to Reuters's satellite transmission, liability should attach whether the software is transmitted by the internet or by CD-ROM.

In any event, the software inventor is protected under copyright law against a competitor who obtains an unauthorized copy of the software in the United States and transmits it abroad for installation on foreign computers. Under *Microsoft v. AT&T*, on the other hand, liability for patent infringement would be foreclosed because the software would not count as a component of a patented device for purposes of § 217(f).[195]

However, the competitor would not be liable under the copyright holdings for transmitting a single *authorized* copy overseas, where it is subsequently

---

[189] Sheldon v. Metro-Goldwyn Pictures Corp., 106 F.2d 45, 52 (2d Cir. 1939) (permitting extraterritorial damages where an infringing movie producer "made the negatives in this country, or had them made here, and shipped them abroad, where the positives were produced and exhibited").

[190] *Id.*

[191] *Reuters*, 149 F.3d at 990-92 (holding Reuters liable for damages resulting from unauthorized transmission to Europe and Africa of Los Angeles News Service helicopter video of the infamous Reginald Denny beating during the 1992 Los Angeles riots).

[192] *Id.* at 990.

[193] *Id.*

[194] *Id.* at 990-92. However, the Ninth Circuit has circumscribed the availability of extraterritorial damages; the copyright owner could not recover damages if the domestic infringer had merely *authorized* the overseas reproduction of the copyrighted work. Subafilms, Ltd. v. MGM-Pathe Commc'ns Co., 24 F.3d 1088, 1091-92 (9th Cir. 1994) (holding that it is not a violation of the Copyright Act to authorize copying completed entirely outside the U.S. because language in the Act reserving to authors the right to "authorize" prohibits only authorizing those acts that by themselves would violate the copyright owner's rights).

[195] *See supra* Part III.A.

reproduced without authorization. In the Second Circuit there would be no predicate act of domestic infringement that creates liability for extraterritorial damages,[196] while in the Ninth Circuit the competitor's actions would amount to merely authorizing overseas infringement, and no liability would attach.[197] If the competitor can circumvent liability by sending an authorized copy overseas, our software inventor would need to turn to international and ultimately foreign law for protection.

The internationalization of copyright law began with the Berne Convention of 1886 and culminated with the Agreement on Trade-Related Aspects of Intellectual Property Rights (TRIPS) of 1994, to which all members of the World Trade Organization (WTO) must adhere.[198] The WTO membership numbers 153 nations, ranging from Albania to Zimbabwe, including Germany.[199] TRIPS expressly protects copyright in computer software as a literary work.[200] The essential tenet of TRIPS is national treatment, the commitment of member states to afford the intellectual property owners of other member states the same protections they give their own nationals.[201] Unlike patent protection, copyright protection under TRIPS requires no application or other formality outside the author's home country.[202] Finally, TRIPS mandates that WTO member nations make copyright protection, but not patent protection, available for software.[203]   Thus, current trends in

---

[196] *See supra* note 194 (describing no liability in the Ninth Circuit where the defendant merely authorized acts of copying that occurred entirely outside the United States.)

[197] *See supra* note 189 (describing liability in the Second Circuit when the overseas acts were predicated upon infringing acts in the U.S.).

[198] *See* Note, *Harmonizing Copyright's Internationalization with Domestic Constitutional Constraints*, 121 HARV. L. REV. 1798, 1802 (2008) (characterizing TRIPS as "the most important intellectual property treaty of the modern era and [one that] substantially ratcheted up minimum intellectual property standards for members of the World Trade Organization").

[199] World Trade Organization, Members and Observers (July 23, 2008), http://www.wto.org/english/thewto_e/whatis_e/tif_e/org6_e.htm (listing the members of the WTO).

[200] Agreement on Trade-Related Aspects of Intellectual Property Rights, art. 10, Apr. 15, 1994, Marrakesh Agreement Establishing the World Trade Organization, Annex 1C, 1869 U.N.T.S. 299, *available at* http://www.wto.org/english/docs_e/legal_e/27-trips.pdf ("Computer programs, whether in source or object code, shall be protected as literary works . . . .").

[201] *Id.* at art. 3 ("Each Member shall accord to the nationals of other Members treatment no less favourable than that it accords to its own nationals with regard to the protection of intellectual property . . . .").

[202] *See* Berne Convention for the Protection of Literary and Artistic Works, art. 5(2), Sept. 6, 1886, 25 U.S.T. 1341, 1161 U.N.T.S. 3 ("The enjoyment and the exercise of these rights shall not be subject to any formality; such enjoyment and such exercise shall be independent of the existence of protection in the country of origin of the work.").

[203] *See* J.H. Reichman, *Universal Minimum Standards of Intellectual Property Protection Under the TRIPS Component of the WTO Agreement*, 29 INT'L LAW. 345, 360-61 (1995)

international harmonization of intellectual property law favor copyright protection for software.

As a result of these provisions, the inventor of the translation software could bring a copyright infringement suit in a German court against a competitor for alleged unauthorized copying in Germany, without having previously registered a copyright there. The inventor would have available the same remedy as a German national, and be assured of at least the minimum standards of protection specified in TRIPS. Indeed, U.S. copyright owners have successfully enforced their copyrights in the courts of WTO members.[204] In a Swedish court, for example, the American entertainment industry received $3.7 million in damages for copyright infringement by the Pirate Bay file sharing site.[205] The U.S. copyright holders went on to seek a Swedish civil court order to shut down Pirate Bay.[206]

Although remedies for copyright infringement are indeed available in overseas courts, the scope of copyright protection may vary from country to country and may fall short of the protection in the United States. However, TRIPS mandates that WTO member nations comply with the requirements of the Berne Convention, and thereby establishes a minimum set of substantive copyright protections.[207] The substantive rights relevant to software are the copyright owner's exclusive rights to reproduce, translate, and adapt.[208] These minimum rights correspond to the exclusive rights to copy and prepare derivative works under U.S. copyright law.[209] However, the Berne Convention omits the guarantee under United States law of the copyright owner's exclusive

---

(referring to "unsettled and controversial" availability of software patent protection in most developed countries to explain the preference for software copyright).

[204] Aaron O. Patrick & Sarah McBride, *Corporate News: Four Guilty in Web Piracy Case*, WALL ST. J., Apr. 18, 2009, at B5 (reporting that the U.S. movie and music industries obtained a criminal complaint against copyright infringers in Sweden).

[205] Patrick & McBride, *supra* note 204 (acknowledging, however, that pressure from the U.S. government was needed to prompt Swedish authorities to take the criminal action that led to the damage award plus a jail sentence for Pirate Bay operators).

[206] Sarah McBride, *Studios Sue to Stop Pirate Bay – Hollywood Seeks Injunction on File-Sharing Site After Earlier Court Win*, WALL ST. J., July 29, 2009, at B7 ("The [major Hollywood] studios are seeking to stop Pirate Bay's operators and its Internet-service provider from making copyrighted movies and TV shows easily available on the Internet.").

[207] Agreement on Trade-Related Aspects of Intellectual Property Rights, *supra* note 200, at art. 9 (requiring members to comply with the Berne Convention).

[208] *See* Berne Convention for the Protection of Literary and Artistic Works, *supra* note 202, at arts. 8-9, 12.

[209] *See* 17 U.S.C. § 106 (2006) ("[T]he owner of copyright under this title has the exclusive rights to do and to authorize any of the following: (1) to reproduce the copyrighted work . . . ; (2) to prepare derivative works based upon the copyrighted work; (3) to distribute copies . . . of the copyrighted work to the public by sale or other transfer of owner-ship, or by rental, lease, or lending . . . .").

right to distribute the copyrighted work.[210] TRIPS partially makes up for this omission in the case of software by requiring WTO members to provide copyright owners the exclusive right to authorize or prohibit software rentals.[211] Thus, between the TRIPS guarantee of copyright protection for software and the exclusive rights of copyright owners to reproduce, translate, adapt, and authorize rental of software, WTO members must afford a uniform level of substantive copyright protection for software that is comparable to the level afforded under United States law.

Protection of patent rights in software, by contrast, varies considerably among nations because substantive patent harmonization has been difficult to achieve.[212] Meaningful harmonization of substantive patent law would require WTO members to "agree to adopt identical rules concerning what constitutes a novel and useful invention, when a technical advance meets the requirement for an 'inventive step' (nonobviousness), and how much information must be revealed by the patent disclosure."[213] Until there is such agreement, national patent law leaves "ample room for national variations and approaches."[214]

Even if the international law of copyrights is relatively better suited than that of patents to protect software-based inventions, additional steps toward global harmonization of intellectual property law will help ensure meaningful protection in the global marketplace. Lax enforcement of existing copyright laws in India accounted for $915 million in lost sales by U.S. companies in 2007 – chiefly due to software piracy.[215] India is a WTO member and therefore bound by the TRIPS agreement,[216] calling into question the agreement's effectiveness. Fortunately, TRIPS incorporates dispute resolution mechanisms through which the United States can work to improve global copyright enforcement, offering measured steps toward better copyright protection for software in cross-border trade.[217]

As part of the Uruguay Round Agreements Act of 1994, the United States adopted a systematic process, known as "Special 301," for improving foreign nations' compliance with TRIPS.[218] Under Special 301, the United States Trade Representative ("USTR") "must identify those countries that deny

---

[210] *See id.*

[211] Agreement on Trade-Related Aspects of Intellectual Property Rights, *supra* note 200, at art. 11.

[212] *See* Reichman & Dreyfuss, *supra* note 182, at 89 (describing how TRIPS left the possibility of a variety of approaches to patent protection).

[213] *Id.* at 90.

[214] *Id.* at 89.

[215] Baldia, *supra* note 168, at 210-11.

[216] World Trade Organization, *supra* note 199.

[217] *See* Reichman, *supra* note 203, at 350.

[218] *See* OFFICE OF THE U.S. TRADE REPRESENTATIVE, 2008 SPECIAL 301 REPORT 18 (2008), *available at* http://www.ustr.gov/assets/Document_Library/Reports_Publications/2008/2008_Special_301_Report/asset_upload_file553_14869.pdf.

adequate and effective protection for [intellectual property rights] . . . ."[219] Violations trigger the USTR to request consultations with the offending nation under the WTO Dispute Settlement Understanding.[220] In 2007, for example, the USTR requested consultations with China over lax Chinese enforcement of copyright anti-piracy laws.[221] When consultations failed to reach a resolution, the USTR escalated the issue by requesting that a WTO panel investigate the United States complaint.[222] Further escalation could result in the WTO assessing damages (known as compensation) against the Chinese, and ultimately permitting U.S. economic retaliation in the form of suspended trade concessions.[223]

Invoking the WTO dispute settlement system to improve TRIPS compliance does not provide direct relief to U.S. software inventors injured by overseas copying. Nevertheless, the USTR credits successful Special 301 actions for improving TRIPS compliance in a long list of nations.[224] Thus, by focusing its attention on the enforcement of software copyright in our trading partners, the USTR could likely improve international copyright protection for software in the long run. This outcome promises to benefit U.S. software innovators in the increasingly important global software market.[225]

In the wake of *Microsoft v. AT&T* and *In re Bilski*, the absence of effective patent protection for U.S. software in the increasingly important international market calls for an approach that does not just reflexively tighten the patent laws. Software inventors, particularly those with a stake in the overseas markets, should focus on litigating their property rights under both United States and increasingly harmonized foreign law. Given the importance of the global trade in software licenses, the USTR should focus on improving enforcement and increasing the scope of copyright protections for software in our trading partners. In the end, this strategy will provide software inventors with their best defense.

---

[219] *Id.*

[220] *See id.* at 16.

[221] *Id.*

[222] *Id.*

[223] *See* Understanding on Rules and Procedures Governing the Settlement of Disputes, art. 3, Final Act Embodying the Results of the Uruguay Round of Multilateral Trade Negotiations, Annex 2, *available at* http://www.wto.org/english/docs_e/legal_e/28-dsu.pdf (setting forth the general provisions of dispute resolution, including provisions for compensation).

[224] OFFICE OF THE U.S. TRADE REPRESENTATIVE, *supra* note 218, at 19-47 (setting forth Watch Lists and discussing progress made in improving TRIPS compliance).

[225] *See supra* Part II.A (discussing the great economic interests at stake in protecting intellectual property rights in software).

CONCLUSION

In the United States, intellectual property rights in software were initially protected under copyright law. However, patents became the preferred means of protecting software as soon as software became widely patentable after the *State Street Bank* decision in 1998. The number of software patents exploded because software inventors were able to claim broad patent rights over the functions performed by the software, without regard to how the software was implemented. Copyright law permits property rights of a narrower scope, tied more closely to the text of the software inventor's implementation. Moreover, patent plaintiffs face a lower burden of proof than copyright plaintiffs. Unfortunately for software patentees, these patent benefits have come to naught in the wake of recent Supreme Court and Federal Circuit decisions that emphasize the strictly territorial scope of patent rights in software. These decisions effectively eliminate U.S. patent protection for software in cross-border trade.

When an American competitor sells an inventor's software without authorization in an increasingly important global market – in some cases the only market – it robs software developers of the incentive to innovate that our system of intellectual property protection exists to create. Software patents might provide a remedy against infringing sales to the domestic market, but not against such sales to a foreign market, as the combined result of *In re Bilski* and *Microsoft v. AT&T* indicates.

These developments suggest that software inventors primarily targeting overseas markets should avoid the expense of obtaining United States and foreign patents, and should instead litigate their rights under U.S. and international copyright law. In contrast to the lack of protection afforded by patents, domestic copyright law provides extraterritorial damages under some circumstances, and international copyright law provides more uniform rights that do not require country-by-country applications. Furthermore, any policy arguments against copyright in software apply with equal or greater force to patents because patents confer stronger property rights that exacerbate their negative effects. To support U.S. software inventors in the global market, the United States Trade Representative should focus its Special 301 efforts on strengthening software copyright protection in our trading partners.

The bell tolls for software patents. By eliminating patent protections for software exports, *In re Bilski* and *Microsoft v. AT&T* caused the demise of patents as an effective means to safeguard property rights in software intended for overseas markets. In their place, software copyrights stand to offer more reliable protection for U.S. software developers who target the global marketplace.