No. 08-964

In The
Supreme Court of the United States

BERNARD L. BILSKI and RAND A. WARSAW,

Petitioners,

v.

DAVID J. KAPPOS, Under Secretary of Commerce
for Intellectual Property and Director,
Patent and Trademark Office,

Respondent.

*On Writ of Certiorari to the United States
Court of Appeals for the Federal Circuit*

**BRIEF OF PROFESSOR LEE A. HOLLAAR AND
IEEE-USA AS *AMICI CURIAE* IN
SUPPORT OF AFFIRMANCE**

DAVID M. BENNION
  Counsel of Record
MICHAEL R. MCCARTHY
PARSONS BEHLE & LATIMER
201 SOUTH MAIN STREET
SALT LAKE CITY, UT 84111
(801) 532-1234

Counsel for Amici Curiae

September 1, 2009

i

Table of Contents

ii

Table of Authorities

Cases

Statutes

Rules

Regulations

Other

Interest of the *Amici Curiae*[1]

Lee A. Hollaar, the author of this brief, is a professor of computer science in the School of Computing at the University of Utah. He teaches courses both in computer and intellectual property law and in computer systems and networking. He has been programming computers since 1964 and designing computer hardware since 1969. He received his B.S. degree in electrical engineering from the Illinois Institute of Technology in 1969 and his Ph.D. in computer science from the University of Illinois at Urbana-Champaign in 1975.

Professor Hollaar is an inventor and patentee of computer-related technology; a Registered Patent Agent involved with the prosecution of patent applications since 1989; an expert witness, consultant, and special master in patent litigation; the author of *Legal Protection of Digital Information* (BNA Books, 2002) and course material on computer-based patents; and a teacher of that material. He is concerned that the decision in this case will continue the unclear lines of what is statutory subject matter by making distinctions not tied to real technological differences or may force inventors of computer-program-based

---

[1] In accordance with Supreme Court Rule 37.6, counsel listed on the cover states that this brief was authored by amicus curiae Professor Hollaar and reviewed by counsel, and that counsel to a party did not author this brief in whole or in part. No person other than the amici curiae and their counsel made a monetary contribution to the preparation or submission of this brief.

Petitioners and Respondent have consented to the filing of all amicus curiae briefs in support of either or neither party and their consents have been filed with the Court.

inventions to claim them in ways that obscure the patentable advance over the prior art.

Dr. Hollaar is the former chair of IEEE-USA's Intellectual Property Committee, where he supervised the filing of the amicus brief whose theory of foreseeability was adopted by this Court in *Festo*,[2] and filed an amicus brief on his own behalf in *Grokster*[3] whose theory of inducement liability was also adopted by this Court.

IEEE-USA is an organizational unit of The Institute of Electrical and Electronics Engineers, Inc. (IEEE), a New York-based not-for-profit, which is the world's leading educational and scientific association for the advancement of technology. IEEE-USA supports the nation's prosperity and competitiveness by fostering technological innovation for the benefit of all, including the more than 210,000 engineers, scientists and allied professionals who are U.S. members of the IEEE.

As part of its mission, IEEE-USA seeks to ensure that U.S. patent and copyright law serves to promote the progress of science and the useful arts consistent with the principles set forth by our nation's founders. IEEE's U.S. members serve on the "front line" of the United States copyright and patent system. Our membership includes inventors and software authors who create and use cutting-edge technology,

---

[2] *Festo Corp. v. Shoketsu Kinzoku Kogyo Kabushiki Co.*, 535 U.S. 722 (2002).

[3] *Metro-Goldwyn-Mayer Studios, Inc. v. Grokster, Ltd.*, 545 U.S. 913 (2005).

who research and publish professional articles and journals, and who develop published standards that form the bases for widely adopted and critical technologies. They are entrepreneurs and employees of firms that acquire, license, and market patented works.

IEEE-USA recognizes that the promotion of progress requires a delicate balancing of these groups interests, and IEEE-USA consistently speaks for that balance. When a decision threatens to disrupt the nation's intellectual property system, IEEE-USA respectfully believes it has the experience and perspective to aid the court as it interprets the law to achieve the constitutional directive of promoting progress. It has done so previously with amicus filings in *Festo* and *Grokster*, suggesting the balanced approach ultimately adopted by this Court.

The Federal Circuit decision in this case, concerning the patenting of method claims, as well as subsequent decisions of the Board of Patent Appeals and Interferences and trial courts, threaten existing and future software patents, an area of law and technology important to IEEE's members.

Summary Of The Argument

This Court's trilogy of software patent cases,[4] and the lower court decisions trying to reconcile them, have led to over three decades of uncertainty. This Court should use this case to repudiate those unfortunate

---

[4] *Gottschalk v. Benson*, 409 U.S. 63 (1972), *Parker v. Flook*, 437 U.S. 584 (1978), and *Diamond v. Diehr*, 450 U.S. 175 (1982).

cases (and thereby the opinions below based on them) and go back to the original clear and simple test for determining when a process is statutory subject matter:

- *A process is patentable subject matter when it involves making or using a machine, manufacture, or composition of matter.*

Such a test is needed so examiner time can be spent more productively on the critical questions of novelty and nonobviousness and not trying to determine if a process is statutory.

The Federal Circuit's opinion in this case just creates new confusion over what it means to be "tied to a particular machine." Already, we are seeing collateral damage to software-based patents, with the Patent Office denying patents and district courts striking down granted patents.

Other proposed tests ("method of doing business," "abstract/applied," "practical application" or "useful, concrete and tangible," or "technological") portend more decades of uncertainty, as the Patent Office Board of Appeals and Interferences and the Federal Circuit try on a case-by-case basis to apply those terms to determine where the boundary really lies.

Judge Dyk's concurrence in this case provides an excellent discussion of how this test was the understanding at the time of the Patent Act of 1793, on which the current statutory language is based, of when a process is statutory subject matter. Although this test is over two centuries old, because the concepts of "machine, manufacture, or composition of matter"

have evolved with technology, so has this test. It provides a good and clear line for even the most cutting-edge technology.

At the same time, this Court should take this opportunity to restate three principles related to that test:

- *Claims drawn to the other categories (machine, manufacture, or composition of matter) are always statutory subject matter.*
- *A claim that covers both statutory and non-statutory embodiments (under the broadest reasonable interpretation of the claim when read in light of the specification and in view of one skilled in the art) embraces subject matter that is not eligible for patent protection and therefore is directed to non-statutory subject matter.*
- *While using a machine may make a claim statutory, if the method itself is known in the prior art, simply adding a machine to perform the method does not necessarily make the claim nonobvious.*

Because the claims at issue in this case are not limited to using or making a machine, manufacture, or composition of matter, they are not statutory subject matter, and the decision below should be affirmed, but on the modified grounds stated above.

Argument

Section 101 of the Patent Act of 1952, 35 U.S.C. § 101, provides the initial coarse filters in determining whether something is patentable. First, any claims must recite statutory subject matter (a "process,

machine, manufacture, or composition of matter, or any new and useful improvement thereof"[5]) and second, the invention must be "useful." It is only after those requirements are met that the novelty requirement of Section 102, 35 U.S.C. § 102, the non-obviousness requirement of Section 103, 35 U.S.C. § 103, and the enablement requirements of Section 112, 35 U.S.C. § 112, need be considered.[6]

Before the advent of computer technology, process claims in patents were used to claim a new way of making or using a known machine, manufacture, or composition of matter, the other classes of statutory subject matter. The first patent granted by the United States, in 1790, was for a new process for making potash. In *Cochran v. Deener*, this Court stated:

> That a process may be patentable, irrespective of the particular form of the instrumentalities used, cannot be disputed. * * * A process is a mode of treatment of certain materials to produce a given result. It is an act, or a series of

---

[5] The brilliance of the original drafters of this provision is that they used general terms which have been able to evolve as technology has evolved, incorporating things that could not have been contemplated two centuries ago, and yet saying that not everything should be patentable. Only one word has been changed ("art" updated to the more common "process") since this provision was enacted in the Patent Act of 1793.

[6] A good summary of the steps that must be passed in order for a patent to be granted on an invention is given by Judge Giles Rich, one of the drafters of the Patent Act of 1952, in *In re Bergy*, 596 F.2d 952, 959-963 (CCPA 1979). The opinion goes on to note some confusion introduced by not tracking the statutory scheme in discussing the patentability of an invention.

acts, performed upon the subject matter to be transformed and reduced to a different state or thing. If new and useful, it is just as patentable as is a piece of machinery.[7]

A trilogy of confusing opinions leads to the question before this Court

The confusion over what is a statutory process began with this Court's decision in *Gottschalk v. Benson*,[8] finding unpatentable a method for "converting signals from binary coded decimal form into binary" claimed as a series of steps. Concluding that Congress should draw the patentability line with respect to software-based inventions,[9] this Court held that the claimed invention was unpatentable. But the opinion also stated "It is said that the decision precludes a patent for any program servicing a

---

[7] 94 U.S. 780, 787-788 (1876), cited in *Diehr*, 450 U.S. at 182-183.

[8] 409 U.S. 63 (1972).

[9] Unfortunately, Congress ignored this Court's suggestion and has said essentially nothing about the patentability of software-based inventions. Congress did adopt a "prior user right" for "method[s] of doing or conducting business" in 1999, see 35 U.S.C. § 273 added by Public Law 106-113 § 1000(a)(9), 113 Stat. 1501A-555, but that appears to be a safe harbor for those who had felt that business methods were unpatentable and had kept their business method as a trade secret so that it was unavailable as prior art. The prior use defense of § 273 prevents a later-issued patent from stopping the continued use of their method. Contrary to petitioner's assertion in the questions presented, there is no indication that Congress ever intended this provision as a statement that business methods were statutory subject matter.

computer. We do not so hold."[10] So, it was left to future cases to determine when a computer program would be patentable.

At the same time, this Court made it clear that the traditional limitation on patentable processes was not absolute, but didn't say where the line is.

> It is argued that a process patent must either be tied to a particular machine or apparatus or must operate to change articles or materials to a "different state or thing."

> We do not hold that no process patent could ever qualify if it did not meet the requirements of our prior precedents.[11]

The two follow-up cases that this Court decided on when a software-based invention was patentable did little to clarify the situation. In *Parker v. Flook*,[12] a claim whose "only novel feature ... is a mathematical formula" was held unpatentable, while three years later, *Diamond v. Diehr*[13] found a method that involved the repeated calculation of a known mathematical formula (the Arrhenius equation) patentable.

---

[10] *Benson*, 409 U.S. at 71.

[11] *Benson*, 409 U.S. at 71.

[12] 437 U.S. 584 (1978).

[13] 450 U.S. 175 (1981).

Having no clear linchpin with which to determine when a software-based invention was patentable or not, the Federal Circuit and its predecessor court, the Court of Customs and Patent Appeals, tried to fashion tests for when a claimed software-based invention is statutory subject matter, as computer technology became more important and conventional machines using gears were replaced by embedded computers in devices such as gas pumps and washing machines.

One only has to look at the multitude of CCPA, Federal Circuit, Board of Patent Appeals and Interferences, and district court opinions, as well as law review articles and briefs in this case, all claiming to be following this Court's trilogy of opinions and yet getting dramatically different results (both Bilski and the Federal Circuit claim fidelity to those opinions), to see that those three opinions have produced nothing but confusion.

It is time for this Court to start with a clean slate and clearly draw the line when a process, especially one implemented using a computer or performing a business method, is statutory subject matter.

The need for a simple test for statutory subject matter

As mentioned previously, whether something is statutory subject matter is just the first of many filters used to determine whether something should be granted a patent. Because patents, unlike any other form of intellectual property protection, don't recognize independent creation as a defense, it is important that only claims that are novel and non-obvious be allowed.

A patent examiner has only a very limited time to determine whether a claim not only meets the threshold requirements of statutory subject matter and usefulness, but also the more substantive requirements of novelty and non-obviousness. Given the limited time an examiner has to process a patent application to either issuance of a patent or a final rejection – on the average, less than twenty hours – it is important that a determination of whether a claim is statutory subject matter be simple to make.[14]

What happened after the Diehr opinion shows how spending time on determining statutory subject matter leads to bad patents

The prosecution history of United States Patent 4,344,142, which is the patent that issued as a result of this Court's decision in *Diehr*, is instructive. Before the appeal, the examiner had rejected the claims only for statutory subject matter and enablement.[15] The

---

[14] Applicants also need a predictable test for statutory subject matter. Since most patent applications are published after eighteen months, see 35 U.S.C. § 122(b), and in as little as six months if a provisional application had been filed, and because it now takes two years or more before an examiner first reviews an application, it is likely that an application will be published (and any chance for trade secret protection lost) before whether a claim's statutory subject matter validity will be assessed.

And since the nature of the claim will determine the form of the disclosure, which can't be changed without losing the application's priority date and possibly the right to any patent because of the application's publication, it is critical that an applicant know at the time of filing that a claim recites statutory subject matter.

[15] The examiner questioned whether undue experimentation would be necessary for a person with ordinary skills to program

examiner had not reviewed the application for novelty or nonobviousness and, in fact, Diehr's attorney tried to goad the examiner on the point by saying that the "applicants infer" that there were no issues in light of the prior art.

Following this Court's decision on March 3, 1981, the application worked its way back to the examiner. On September 28, 1981, Diehr slightly amended the claims so that they better matched the way the courts had interpreted them.[16] On March 23, 1982, the examiner allowed the application as amended.

Rather than a normal office action where the examiner cites prior art and indicates that the claimed invention would be obvious in light of that prior art, the office action reads more like a response from an applicant. Two patents are cited as prior art, and the examiner then points out how each patent differs from the claimed invention. The examiner does not discuss why those differences would not be obvious,

---

the computer to control "a plurality of rubber-molding presses simultaneously." The applicant eventually overcame that rejection by submitting affidavits regarding how it would be possible to write the program based on the specification.

[16] Of significance, "repetitively calculating in the computer, at frequent intervals during each cure, the Arrhenius equation" was changed to "repetitively performing in the computer, at frequent intervals during each cure, integrations to calculate from the series of temperature determinations the Arrhenius equation" and "indicates equivalence" was changed to "indicates completion of curing."

particularly in light of other prior art that may teach those differences.[17]

Since the patent is essentially on the implementation on a computer of a well-known equation first proposed in 1884, one would have expected at least some examination for obviousness by the examiner, but the record shows none. This is especially true after Diehr had submitted affidavits saying how straightforward the programming of a computer to implement the claimed method would be, even for controlling a number of molds simultaneously.

In reviewing other prosecution histories of early software-based patents,[18] when the examiner initially rejected the claims on the grounds that they didn't recite statutory subject matter, much of the prosecution of the application was spent finding claim language that would get around the examiner's initial statutory subject matter rejection. When such language was finally found (and it often was not substantially different in scope from the original claim

---

[17] One difference was that Diehr explicitly claimed determining the temperature "at a location closely adjacent to the mold cavity" and the other difference was that Diehr performed a continuous comparison.

[18] On June 2, 1995, the United States Patent and Trademark Office proposed examiner guidelines for determining when a software-based invention was statutory subject matter, with final guidelines effective on February 28, 1996. In light of the guidelines, rejections based on lack of statutory subject matter substantially declined, both because applicants had a clear idea of how to claim their software-based inventions to meet the statutory subject requirement and because examiners had a clearly-stated test for determining statutory subject matter.

language), the application was allowed with only the slightest consideration of prior art.

Whatever this Court says regarding when a claim written as a series of steps recites statutory subject matter, it is vitally important to the patent system that the test be clear and straightforward to apply. Otherwise, too much time will be spent on applying the test to the detriment of the more important determination of whether a claimed invention is really a non-obvious contribution deserving patent protection.

Where to draw the line

The Federal Circuit's method patentability case law, particularly regarding software-based inventions and business methods, has been confusing and controversial because the Federal Circuit has been bound by this Court's trilogy of software patent cases, which do not draw a clear patentability line for methods. Fortunately, this Court can start with a clean slate simply by overruling any aspect of the trilogy that is inconsistent with this case. Patent examiners and trial courts will no longer have to try to reconcile the irreconcilable, making distinctions that are not anchored in reality.

The Federal Circuit's opinion just creates new confusion, with software-based inventions as collateral damage

The Federal Circuit opinion in this case shows the confusion caused by this Court's software-patent trilogy. In trying to reconcile the language of those three opinions (and overruling its past opinions where

it felt there was a departure from those three opinions), the Federal Circuit held that a method is statutory if "it is tied to a particular machine or apparatus, or transforms a particular article into a different state or thing,"[19] parroting *Benson*'s "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."[20]

What the Federal Circuit did in this case with regard to whether a software-based invention claimed as a method recites statutory subject matter is just kick the can down the road.

Is a general-purpose digital computer a "particular machine or apparatus" or is something more required?[21] The Patent Office's Board of Patent Appeals and Interferences apparently thinks so. In an "informative opinion" last year, the Board found that a claimed method that is "not tied to 'a particular machine,' but rather is tied only to a general purpose computer" is not patentable. It is not clear what characteristics of a computer need to be specified to make it a "particular machine," especially "because a general purpose computer in effect becomes a special

---

[19] *In re Bilski*, 545 F.3d 943, 954 (Fed. Cir. 2008).

[20] *Benson*, 409 U.S. at 70.

[21] *Ex parte Langemyr*, Appeal 2008-1495 at 24, 89 USPQ2d 1988 (BPAI, 2008). Perhaps the Board held the way that it did in hopes that the applicant would appeal to the Federal Circuit and have that court say what it means by "a particular machine or apparatus."

purpose computer once it is programmed to perform particular functions pursuant to instructions from program software."[22]

And at least two district courts have invalidated patents whose claims were directed to a "computer aided method."[23] Tens-of-thousands of patents that claimed software-based inventions as methods are now open to question.

Is the requirement of a particular component necessary for performing the claimed method enough? The claim in *Benson*, found not to be statutory subject matter by this Court, is directed to a particular machine or apparatus – an electrical circuit that includes a "reentrant shift register."[24] Would it be

---

[22] *In re Alappat*, 33 F.3d 1526, 1545 (Fed. Cir. 1994), citing *In re Freeman,* 573 F.2d 1237, 1247 n.11 (CCPA 1978); *In re Noll,* 545 F.2d 141, 148 (CCPA 1976); and *In re Prater,* 415 F.2d 1393, 1403 n.29 (CCPA 1969).

[23] *Cybersource v. Retail Decisions*, No. C 04-03268 MHP, 2009 U.S. Dist. LEXIS 26056 (N.D. Cal. Mar. 26, 2009) and *DealerTrack v. Huber*, No. CV 06-2335 AG, 2009 U.S. Dist. LEXIS 58125 (N.D. Cal. Jul. 7, 2009).

[24] A shift register is a hardware device that shifts all the bits stored in it one position, either left or right, depending on the type of shift register or the control signal to the shift register. A reentrant shift register takes the bit shifted out from the end of the shift register and stuffs it back in the other end.
    At oral arguments, "the Patent Office Solicitor admitted that the reference to this piece of apparatus in the claim was, for him, its 'most embarrassing phrase.'" The CCPA noted that it was not only a phrase, but also a key part of a number of claim elements, indicating that claim 8 only claimed the technique in the context

patentable under the Federal Circuit's opinion in this case because the method is tied to a particular machine or apparatus?

It is likely that unless this Court makes a clear and unambiguous statement regarding when a method is statutory subject matter, applicants (and their attorneys, who are often engineers just as creative as their clients) will continue to push the boundaries of patentability. Affirmation by this Court just means that Bilski has to figure out a different form for his claims, based on some other quote taken out of context in this or a subsequent opinion.

The proposed alternatives will lead to decades of litigation trying to determine where the line lies

A number of alternatives to the Federal Circuit's "transformation or machine" test have been proposed. Some have suggested that "methods of doing business" not be patentable, reversing *State Street Bank v. Signature Financial*,[25] ignoring that the test was "poorly defined, redundant, and unnecessary" in light of the other statutory tests.[26] Bilski and others instead propose that this Court restore their view of *State Street Bank* – that any process is patentable as long as it produces a "useful, concrete and tangible result,"

---

of a particular machine or apparatus. *In re Benson*, 441 F.2d 682, 687 (CCPA 1971).

[25] 149 F.3d 1368 (Fed. Cir. 1998).

[26] *In re Schrader*, 22 F.3d 290, 298 (Fed. Cir. 1994, Newman, J. dissenting).

ignoring that that statement was in the context of a claimed machine. Still others propose that to be patentable something must make a "technological contribution." Judge Rader, in his dissent below, and others say that the test should be whether what is claimed is an "abstract idea."

Adoption of any of these approaches portends decades more litigation as applicants probe the boundaries of patentability and alleged-infringers try to invalidate issued patents. Much as the Federal Circuit, in the opinion below, simply replaced the question "what is a process" with the questions "what is a particular machine, and what does it mean to be tied to it," all of the tests mentioned above simply raise new definitional questions.

For example, those opposed to patents on software-based inventions (or an infringer faced with a software-based patent and no other defense) will argue that software is "abstract." Given the fact that many improvements to machines today are the result of the software on the embedded processor that controls the machine, an opinion holding software abstract would take a major area of innovation out of the patent system.

Software-based inventions need and deserve patent protection as processes

The use of computer technology has expanded well beyond the use of an expensive digital computer to control an industrial process when this Court last considered the patentability of software-based inventions more than a quarter-century ago to being seemingly omnipresent. Microwave ovens, washers

and dryers, television sets and radios, thermostats, furnaces and boilers, sprinkler controllers, and clocks and watches are but a few of the appliances where an embedded computer has replaced mechanical timers, gears, and switches, resulting in more reliable products able to perform more functions at lower prices.

Many of the new capabilities for such appliances are the result of novel and nonobvious computer programs that control traditional machines in ways that cannot be realistically done using mechanical controls. Much as Diehr's invention sensed the actual tire mold temperature and determined what to do next, today's microwave ovens sense the temperature of meat being cooked and dryers sense the moisture content of the clothes.

But many innovations are not limited to a particular type of appliance, but are advances that can benefit a wide range of computer programs. Although a new way to convert BCD numbers to binary is not that important, a technique for sorting information much faster would be because sorting is a fundamental activity in database systems, word processors (to handle things like indexes), compilers, and many other types of computer programs.

But the method in *Benson* wasn't patentable *because* it had a wide range of possible uses.

> The mathematical formula involved here has no substantial practical application except in connection with a digital computer, which means that if the judgment below is affirmed, the patent would wholly pre-empt the

mathematical formula and in practical effect would be a patent on the algorithm itself."[27]

That is a particularly perverse observation with respect to patents, because it denies patent protection to new techniques that have many applications, while presumably allowing patents on techniques that have limited applicability.

One argument the opponents of patents for software-based inventions make is that software is mathematics, and mathematics is not patentable.[28] But that not only ignores how software-based inventions are generally claimed, but is bad computer science. Claims that include data structures in random-access memories, input devices such as keyboards or mice, screen display devices, or clocks

---

[27] *Benson*, 409 U.S. at 71-72.

[28] *Benson*, 409 U.S. 63, (1972) is generally cited for the latter proposition. But in most instances, the correspondence between computer programs and mathematics is merely cosmetic. For example, the equation $E = MC^2$ expresses a relationship between energy and matter first noted by Einstein, while the computer program statement $E = M * C ** 2$ represents the calculation of M time C raised to the second power and then assigning the result to a storage location named E. It is unfortunate for purposes here that the early developers of programming language made their calculation-and-assignment statements look like mathematical equations so that they would seem familiar to scientists and engineers. But the common programming statement $I = I + 1$, which increments the value stored in location I, is essentially nonsense as a mathematical equation. Similarly, a computer program is a series of calculation-and-assignment statements that are processed sequentially, not a set of simultaneous mathematical equations that are solved for their variables.

and time-outs, common in software-based patents, are no longer equivalent to pure mathematics.

Other protections, such as copyright or trade secret, are not sufficient and employing them will only result in distortions in their law

Although computer software can be protected by copyright, that is not sufficient since copyright protection does not "extend to any idea, procedure, process, system, [or] method of operation,"[29] the very innovation that patents are intended to protect. Instead, today it is essentially limited to protection against literal copying or taking advantage of source code from a former employer. But before patents on software-based inventions became common, courts were expanding the nonliteral scope of copyright to "structure, sequence, and organization" of computer programs,[30] giving patent-like protection without the benefit to the public of disclosure and claiming.

Since patents have become generally accepted for protecting innovative techniques in computer programs, copyright protection for computer programs has been scaled back by the courts so that it covers literal copying – buying a single copy of a commercial software product such as Microsoft Windows or Office and installing it on every machine a dealer sells – or when a former employee or other person with access to

---

[29] 17 U.S.C. § 102(b).

[30] *Whelan Associates Inc. v. Jaslow Dental Laboratory, Inc., et al*, 797 F.2d 1222 (3d Cir. 1986). This case essentially provided patent-like protection to software without the requirements of disclosure and clear claiming.

a program's source code copies not only the ideas in the program (which could be trade secret misappropriation if they are not patented) but also the way those ideas have been implemented ("expressed").

Any reduction in the availability of patent protection for software-based techniques would likely result in new distortions to copyright as it is pushed to protect ideas in addition to expression, much as it was in *Whelan*'s time.

And in many cases, a new technique is self-revealing, so that trade secret protection is not available.[31]

---

[31] Many programmers forego trade secret protection for their source code by making it available under a "free" or "open-source" license. These licenses, which are backed by copyrights of the source code, often require anybody improving the program to make their modifications available under the same license. But that does not keep someone from taking any new ideas in the program and incorporating them in a program while keeping their source code as a trade secret, because copyright does not protect ideas.

Although most open-source programmers are opposed to software patents, this likely stems from a time when many of their programs were "clones" of existing proprietary programs (the original "free software" program was to be a reimplementation of the Unix operating system), and a patent could prevent such cloning.

In this respect, they are like developing countries, which downplay intellectual property protection until they start producing their own innovations, at which time copyrights and patents become more interesting. The United States did not protect the works of foreign authors until 1891, when protection of American authors' works in foreign countries became important.

The use of method claims for software-based inventions results in clearer claiming

The question is then how such software-based inventions should be claimed, particularly to meet the statutory requirement that:

> The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.[32]

One way is as a machine or apparatus, which avoids the question of what processes are patentable.[33] This is what was done, for example, in the first claim of an early patent for a software-based invention:[34]

> 1. Apparatus for detecting matches between strings of information-representing signals comprising:
> [a] means for comparing each subunit of a first string to the first subunit of a second string;

---

[32] 35 U.S.C. § 112, second paragraph.

[33] Unless the machine claims are somehow treated as process claims, as was done by the trial court in *State Street Bank* and recently in *Research Corporation Technologies v. Microsoft* (No. CV-01-658-TUC-RCJ, 2009 U.S. Dist. LEXIS 71883, D. Az, July 28, 2009). It might have been far better if the Federal Circuit had simply held that the trial court was wrong to treat as a process what was clearly claimed as a statutory machine.

[34] Kenneth L. Thompson, "Text Matching Algorithm," United States Patent 3,568,156, granted March 2, 1971, and assigned to Bell Telephone Laboratories. Ken Thompson is one of the principal creators of the Unix operating system.

[b] means for recording the identification of that subunit of the second string which subunit follows each matched subunit of the first string;
[c] means for comparing each identified subunit of the second string to the next succeeding subunit of the first string;
[d] means for indicating a successful match when all subunits of the second string are compared; and
[e] means for indicating an unsuccessful match when all subunits of the first string are compared.

Since the elements of the claimed apparatus are in means-plus-function form, we have to look to the specification to see what is really being claimed.[35] And what we find is two very different implementations of the claimed special-purpose machine – the first in conventional digital logic of the period (DEC "Flip Chip" modules) and the second as assembly-language program segments for the IBM 7094 computer.[36]

---

[35] See 35 U.S.C. § 112, sixth paragraph:

An element in a claim for a combination may be expressed as a means or step for performing a specified function without the recital of structure, material, or acts in support thereof, and such claim shall be construed to cover the corresponding structure, material, or acts described in the specification and equivalents thereof.

[36] Presumably it was claimed this way to make the examiner see statutory subject matter (at least the digital logic) and then leverage that into a patent on the software implementation, both through claim 1 and a corresponding method claim 2. This application was filed at the time that Bell Labs was trying to determine the patentability of computer-based inventions. Another application that they filed became the subject of

Such a claim is undesirable because it is not clear on its face what is covered by the patent. One has to read through the specification and guess what structure corresponds to each functional element[37] and what may be equivalent to the structure in the specification. The Federal Circuit, in *State Street Bank*, apparently felt that it was necessary to include its view of the means for implementing each claim element so that the claim could be understood.[38]

A recent book posits that a major problem with patents today is that it is difficult for people to determine what is covered by a patent, and this lack of

---

Gottschalk v. Benson et al., 409 U.S. 63 (1972).

Similarly, when the invention of *State Street Bank* was claimed both as a method and a machine, the examiner rejected the method claims but allowed the corresponding machine claims. 149 F.3d at 1371.

[37] It is not necessary for the applicant to indicate the corresponding structure in the specification for a claim element to the examiner, and examiners seldom indicate what they considered the corresponding structure. The Board of Patent Appeals and Interferences does require that:

> every means plus function and step plus function as permitted by 35 U.S.C. 112, sixth paragraph, must be identified and the structure, material, or acts described in the specification as corresponding to each claimed function must be set forth with reference to the specification by page and line number, and to the drawing, if any, by reference characters.

37 C.F.R. § 41.37(c)(1)(v). Because of that, every appeal coming to this Court from the Board will have the corresponding structure identified. But that is not the case for patents that have not been appealed to the Board.

[38] 149 F.3d at 1371-1372.

a predictable property right produces uncertainty for developers and costly disputes that may outweigh the positive incentives of the patent system.[39] The authors' research found that only in some sectors of technology, such as the pharmaceutical industry, do patents act as advertised, with their benefits outweighing their costs, while for software, the lack of clear claiming has a definite negative effect.

Software-based inventions can be claimed in a more straightforward manner using method claims, where the elements of the claims are the steps of the method performed by the new software technique.[40] This way of claiming simplifies the examination of the application, because it concentrates on the new method that is the heart of the invention. Much innovation today is in the techniques implemented using a computer, such as a more sophisticated way of controlling energy usage by an appliance which could

---

[39] James Bessen and Michael J. Meurer, *Patent Failure: How Judges, Bureaucrats, and Lawyers Put Innovation at Risk*, Princeton University Press, 2008, at 46-72 ("If You Can't Tell the Boundaries, Then It Ain't Property").

[40] Some people may try to justify such a claim by saying that the claimed method "transforms" a general-purpose computer into a "special-purpose" computer. But that ignores when the claimed method occurs, which is when the "transformed" computer performs all the steps of the method. A method claim for transforming the computer would have steps such as "loading the program implementing the method from a storage device into the instruction-storage memory of the computer, so that the instructions implementing the steps of the method can be executed." Such a method is clearly obvious in light of decades of loading program instructions to be executed into memory.

not be practically implemented using a mechanical controller.

For example, the claim above rewritten as a method claim would be:

> The method of detecting matches between strings of electronically coded subunits comprising the steps of:
> [a] comparing each subunit of a first string to the first subunit of a second string;
> [b] recording the identification of that subunit of the second string following each matched subunit of the first string;
> [c] comparing each identified subunit of the second string to the next succeeding subunit of the first string;
> [d] indicating a successful match when all subunits of the second string are compared; and
> [e] indicating an unsuccessful match when all subunits of the first string are compared.[41]

Not only does the use of the process claim not require that the specification be consulted to find corresponding structure to determine the scope of the claim, but it is infringed not when some collection of parts capable of performing the claimed method happen to come together in one machine, perhaps from

---

[41] This is, in fact, Claim 2 of the 3,568,156 patent, except that the steps of the method are numbered rather than using letters to correspond to the "means-for" elements of the apparatus claim.

different programs, but when the new method is actually performed.[42]

It should not be surprising that it is convenient to claim the technique of a software-based invention using language that is similar to describing the steps a person would do to perform the technique. But that is not because it is practical for a person to actually perform the technique, but because of our tendency to anthropomorphize computers,[43] describing the things done by a computer as if it were being done by a person.

But a claim to a software-based invention will often contain language explicitly limiting it to a computer (often in the preamble) or implicitly because of the nature of claim elements. For example, in the claim immediately above, the preamble limits the claim to

---

[42] For those wanting to catch people constructing the special-purpose computer created by the method, a dependent claim such as "A digital computer system programmed to perform the method of claim N" can be used. Similarly, an article of manufacture claim to catch those producing the disks used to distribute an implementation of the method, a claim like "A computer-readable medium storing a computer program implementing the method of claim N" can be used. With both, the examiner can concentrate on determining the novelty and nonobviousness of the claimed method, and then simply check to see that the dependent claims are in the proper form.

[43] Ironically, the term "computer" referred to a person who carried out calculations or computations, such as those employed by the military to calculate ballistics tables. As analog and digital computers took over these tasks, in the middle of the twentieth century, the term took on today's meaning of a device performing the computations.

"electronically coded subunits" and not strings of characters written on paper to be read by a person.

Claiming a software-based invention as a method is clearly not the type of clever claiming this Court warned about when it said:

> A competent draftsman could attach some form of post-solution activity to almost any mathematical formula; the Pythagorean Theorem would not have been patentable, or partially patentable, because a patent application contained a final step indicating that the formula, when solved, could be usefully applied to existing surveying techniques. The concept of patentable subject matter under §101 is not "like a nose of wax which may be turned and twisted in any direction * * *."[44]

Instead, claiming a software-based invention as a method makes it clear where the novelty of the invention rests, which in turn makes it easier for the examiner to assess whether the requirements of novelty and non-obviousness have been met.

Using a process claim rather than a machine claim,[45] coupled with the Federal Circuit's developing

---

[44] *Parker v. Flook*, 437 U.S. 584, 590 (1978).

[45] If it is desirable to also claim the software-based invention as a machine or apparatus for purposes of who infringes, that can easily be done with a dependent claim in the form of "A digital computer system programmed to perform the method of [the parent method claim]."

law on full-scope enablement,[46] applicants will specify in the claim the particular steps of their claimed method that creates the special-purpose computer, but not use language overly broad lest their patent be invalid for lack of enablement. Unlike claiming the invention as a machine using functional elements, there will be no need for a person wanting to know the scope of the claims to guess at what structure in the specification defines each claim element, how broadly that structure should be read, and what are its equivalents.

This Court should support the use of process claims for software-based inventions because they represent a way of more clearly claiming what would otherwise be statutory: the special-purpose machine programmed to perform the method.

A clear line: statutory methods make or use a machine, manufacture, or composition of matter

Judge Dyk's concurrence below notes that the original understanding at the time of the first patent statutes was that an "art" in the patent sense (changed to "process" by the Patent Act of 1952) involved "working or making of any manner of new manufacture," and that "manufacture" – something

---

[46] See, for example, *Sitrick v. Dreamworks*, 516 F.3d 993 (Fed. Cir. 2008). It is far, far better for examiners to spend their limited time not only determining whether a claimed invention is really novel and non-obvious as well as making sure that the scope of the claim is commensurate with what has been disclosed, than to try to apply confusing and sometimes contradictory rules to determine whether a claim recites statutory subject matter.

man-made – encompassed "machines" and "compositions of matter."[47]

This comes from the English Statute of Monopolies, which was the basis of United States patent law, and Congress explicitly recognized that limit on statutory subject matter by including similar categories in the early patent acts and carrying them over to the present statutes.[48]

Judge Dyk also notes that "methods of organizing human activities" were never considered patentable at the time the patent statutes were written, even though they clearly existed at that time.[49]

---

[47] *In re Nuijten*, 515 F.3d 1361 (Fed. Cir. 2008, *cert. denied*), held that even though signals are in a sense "man-made," they are not "manufactures" as that term is used in Section 101. 515 F.3d at 1356-1357.

[48] The first book published in the United States on patent law notes that the language of the Statute of Monopolies and the Patent Act of 1793 are "co-extensive" and "British authorities, therefore, so far as circumstances are similar, and the reasons for the cases remains the same, will be pertinent for determining what kinds of new invented things may be the legal subjects of patent monopolies." Thomas G. Fessenden, *An Essay on the Law of Patents for New Inventions* (1810), at 59. (Available at http://www.ipmall.info/hosted_resources/ip_antique_library/Patent/Fessenden_1810.pdf)

[49] It is possible that some patents have been granted for processes that seem not to meet this test, but in some instances that may be because one is looking at the title of the patent and not what was actually claimed. In other instances, the patent may have been granted without full consideration of the rule.

Updating the Statute of Monopolies' formulation to use current terms from the patent statutes, we get a clear statement of when a process is patentable: *A process is patentable subject matter when it involves making or using a machine, manufacture, or composition of matter.*

Although this test is over two centuries old, because the understanding "machine, manufacture, or composition of matter" has evolved with technology, so has this test. It provides a good and clear line for even the most cutting-edge technology.

With respect to "using," that means that at least one step of the claimed process is performed by a specified machine or the person performing the method uses a specified machine, manufacture, or composition of matter in order to perform at least one step of the claimed process.

Software-based inventions would be statutory under this test as a method if they are claimed as running on a digital computer, which would clearly be "using a machine." The method claims would not be for "making a machine," since as noted previously, such a claim would have steps such as "loading the program implementing the method from a storage device into the instruction-storage memory of the computer, so that the instructions implementing the steps of the method can be executed," and would most likely be obvious in light of past techniques for loading programs into a digital computer to be run.

As noted above, sometimes courts have applied the rule for whether a process is patentable to machines, manufactures, and compositions of matter. It is worth

restating the rule that: *Claims drawn to the other categories (machine, manufacture, or composition of matter) are always statutory subject matter.* Machines, manufactures, and compositions of matter are all concrete things. They will never be "laws of nature, physical phenomena and abstract ideas,"[50] the exceptions to statutory subject matter.

There is another rule that needs restating: *A claim that covers both statutory and non-statutory embodiments (under the broadest reasonable interpretation of the claim when read in light of the specification and in view of one skilled in the art) embraces subject matter that is not eligible for patent protection and therefore is directed to non-statutory subject matter.*[51]

This means that in the broadest reading of the claim,[52] it must involve making or using a machine, manufacture, or composition of matter. As an example, consider the two claims in *Benson*.[53] Claim 8 was:

---

[50] *Diehr*, 450 U.S. at 185.

[51] United States Patent and Trademark Office, "Interim Examination Instructions for Evaluating Subject Matter Eligibility Under 35 U.S.C. § 101," August 2009.

[52] "[A]s an initial matter, the PTO applies to the verbiage of the proposed claims the broadest reasonable meaning of the words in their ordinary usage as they would be understood by one of ordinary skill in the art, taking into account whatever enlightenment by way of definitions or otherwise that may be afforded by the written description contained in the applicant's specification." *In re Morris*, 127 F.3d 1048, 1054 (Fed. Cir. 1997).

[53] See 409 U.S. at 73-74.

The method of converting signals from binary coded decimal form into binary which comprises the steps of –
(1) storing the binary coded decimal signals in a reentrant shift register,
(2) shifting the signals to the right by at least three places, until there is a binary '1' in the second position of said register,
(3) masking out said binary '1' in said second position of said register,
(4) adding a binary '1' to the first position of said register,
(5) shifting the signals to the left by two positions,
(6) adding a '1' to said first position, and
(7) shifting the signals to the right by at least three positions in preparation for a succeeding binary '1' in the second position of said register.

Because the claimed process uses a machine or manufacture (the "reentrant shift register"), the claim is statutory. In this respect, the opinion of this Court in this case would overrule *Benson*. In contrast, Claim 13 was:

A data processing method for converting binary coded decimal number representations into binary number representations comprising the steps of –
(1) testing each binary digit position i, beginning with the least significant binary digit position, of the most significant decimal digit representation for a binary '0' or a binary '1';
(2) if a binary '0' is detected, repeating step (1) for the next least significant binary digit position of said most significant decimal digit representation;

(3) if a binary '1' is detected, adding a binary '1'at the (i+1)th and (i+3)th least significant binary digit positions of the next lesser significant decimal digit representation, and repeating step (1) for the next least significant binary digit position of said most significant decimal digit representation;

(4) upon exhausting the binary digit positions of said most significant decimal digit representation, repeating steps (1) through (3) for the next lesser significant decimal digit representation as modified by the previous execution of steps (1) through (3); and

(5) repeating steps (1) through (4) until the second least significant decimal digit representation has been so processed.

Because the claim is not limited to the making or using of a machine or manufacture, and instead can be performed by a person, it would not be statutory. Similarly, the claim in this case would not be statutory because it is not limited to the making or using of a machine, manufacture, or composition of matter.

This new test could be satisfied by an applicant by simply reciting in the claim's preamble that the method operates on, or uses, a digital computer system. But another principle should be make clear: *While using a machine may make a claim statutory, if the method itself is known in the prior art, simply adding a machine to perform the method does not necessarily make the claim nonobvious.*[54]

---

[54] See 35 U.S.C. § 103.

Even if the claim does not explicitly recite a machine, it may be statutory if it recites elements that require a machine. So, for example, limiting a claimed method to a "relational database management system" would also make it statutory, since such a method can only be performed on a machine with such software.[55]

The claims in *Diehr* would be statutory subject matter, but whether they are patentable would depend on whether their method was nonobvious in light of its use of a well-known equation. Novelty, nonobviousness, and enablement should be the primary tests for patentability, not some long debate with the examiner and the courts over whether the claim recites a "particular" machine after which the examiner "throws in the towel" and grants the patent.

Conclusion

There is no indication that Congress intended that anything that can be described as a series of steps should be entitled to patent protection. Yet for software-based inventions, claiming by describing the particular steps of a method provides the clearest notice of what the inventor regards as his invention.

The question is where to draw the line. It is important that such a line be drawn to correspond

---

[55] In *Ex parte Koo and Leung*, Appeal 2008-1344 (BPAI, November 26, 2008), the Board found that such a claim was *not* statutory because the system of the claim "could be a software system" and not a particular machine as the Federal Circuit required in its opinion in this case. That simply underscores the problem with the Federal Circuit's test, because of course a "software system" will perform the steps of the method only when run on a machine.

with reality, or problems will result in applying the test. If the test for a statutory process is simply that it produces a "useful, concrete, and tangible result," it would open the floodgates to patents on just about anything done by man. Tests like the Federal Circuit's "transformation or machine," or whether the process is "abstract" or "technological" or a "business method" would mean years and years of litigation to determine where those vague terms actually draw the line.

Instead, the test should be the one that has existed from the start of patent law, updated to use today's statutory terms: ***A process is statutory subject matter when it involves making or using a machine, manufacture, or composition of matter***.

Respectfully submitted,

David M. Bennion
  Counsel of Record
Michael R. McCarthy
Parsons Behle & Latimer
201 South Main Street
Salt Lake City, UT 84111
(801) 532-1234

Counsel for Amici Curiae
Professor Lee A. Hollaar
and IEEE-USA