THE UNITED STATES MOVES AHEAD OF THE EUROPEAN UNION IN PATENT PROTECTION FOR COMPUTER SOFTWARE

INTRODUCTION

Patent protection is much sought after by persons or companies who have invented a valuable machine or process. The limited monopoly a patent provides gives exclusive rights to control that invention. Not all inventions or new ideas are patentable, however. Patents directly on computer programs have been historically unavailable in the United States and Europe. Software has long been considered non-statutory subject matter. Therefore, copyright has been the popular alternative to patent protection, but it is not as strong as a patent. Copyright does not protect the idea behind the software, and that idea or invention is what many developers want to protect. On June 2, 1995, the United States Patent and Trademark Office (PTO) issued "Request for Comments on Proposed Examination Guidelines for Computer-Implemented Inventions." This publication signals a change in attitude by the PTO toward computer software patents. The patent systems of all industrialized nations, including the United States, were established for the purpose of stimulating technical progress. This stimulus is in the form of a limited monopoly, giving the patent holder exclusive rights to his ideas. These exclusive rights are an economic reward for inventing new and useful technologies. Computer programs are as deserving of the protection of patent laws as any other technology. The new patentability of computer programs gives American software developers more incentive to invent than their European Union counterparts.

The purpose of this note is to discuss the differences in software patent protection in the European Union and the United States. Part II of this note discusses the relative merits of software patents, including the views of the United States government and software industry, as well as the European view. In Part III of this note, the statutory and case law development of the United States is studied. Part IV discusses European treaties and case law, while Part V analyzes the United States Patent Office's Proposed Guidelines and shows how the guidelines conform to case precedent.

I. THE PATENT SYSTEM

A grant of a patent by the United States government confers upon an individual the right to exercise a monopoly over his invention.[1] This

---

1. The right granted by statute is the right to exclude others from making, using, or selling the patented invention. THE CONCISE GUIDE TO PATENTS TRADEMARKS AND COPYRIGHTS 6 (Solomon J. Schepps ed., 1980).

monopoly is given to an inventor as an incentive to invent new and useful products. In exchange for a limited monopoly on his idea, the inventor provides a new and useful invention for society to use when the term of his patent expires. The category of patentable subject matter[2] is very broad, but it does not include natural principles,[3] such as the mathematical formula $E=mc^2$. The principles of nature have always existed and are commonly considered to be in the public domain.[4] An item which is in the public domain cannot be removed from the public domain by a patent. Although Webster's Dictionary defines "invent" as "to discover,"[5] invention is commonly viewed as an application of a scientific principle to solve a problem. The distinction is that a person can "discover" a scientific principle but cannot take it out of the public domain, while an inventor must create something new and unique in order to "invent." The production of a new and unique idea is the very basis of the patent system in the United States and around the world.

A United States patent is comprised of two parts. The first part contains a written description of the invention, which may include the problem that the invention addresses, a description of the prior art in the field, and a summary of the advantages of the invention followed by a detailed technical description of the invention.[6] The second part of a patent is a claim or set of claims which "particularly point out" the subject matter of the invention.[7] The claims define the idea that is being patented and delineate the scope of the patent.

There are several statutory conditions which must be met before a patent will be issued. These conditions include: inventorship, priority over any other similar patent, novelty, lack of publication or prior patenting within one year, non-obviousness, and statutory subject matter.[8] This last condition is the focus of this note as applied to computer software.

A. *Software Industry Performance*

The international software industry is a swiftly growing industry. The United States Department of Commerce puts 1994 worldwide sales of

---

2. Statutory subject matter is the definition by statute of the classes of inventions that are included in the patent law. TOM ARNOLD & FRANK S. VADEN III, INVENTION PROTECTION FOR PRACTICING ENGINEERS 21-22 (1971).

3. ROBERT A. CHOATE & WILLIAM H. FRANCIS, CASES AND MATERIALS ON PATENT LAW 471-72 (1981).

4. *Id.* at 471-74.

5. WEBSTER'S NEW UNIVERSAL UNABRIDGED DICTIONARY 1622 (Deluxe 2d ed., Dorset & Baber 1983).

6. CHOATE & FRANCIS, *supra* note 3, at 85.

7. *Id.* at 85-86.

8. *See* 35 U.S.C. §§ 101-112 (1995).

packaged software at more than seventy-seven billion dollars.[9] The European software market will expand at a slower pace than other regions, reaching forty-five billion dollars in the year 2000.[10] In contrast, the computer software industry is the fastest growing industry in the United States.[11] The U.S. software industry grew by 269 percent in the years 1984-1994, while the total U.S. economy grew by only thirty percent.[12] The International Data Corporation predicts that the United States software market will grow at a rate of thirteen percent, from forty billion dollars in 1995 to seventy-four billion dollars in the year 2000.[13] The strength of the American software industry is demonstrated by a seventy-five percent share of the world market for prepackaged software.[14] The growth in the software industry is reflected in the number of software patents granted. In America in 1989, more than 700 software patents were granted.[15] In 1993, the total number of software-related patent applications had grown to 8391, of which 3613 were actually granted.[16] It is estimated that around the globe there are 30,000 mainframe computers, 300,000 super minicomputers, and nearly 100 million personal computers.[17] These numbers demonstrate that the domestic software industry is of vital importance to the American economy and will continue to grow in importance. Protecting and encouraging this industry will be necessary for America's future economic health.

B. *United States Software Developers Will Benefit from the New Guidelines*

     The new attitude by the PTO will give United States software developers an advantage over their European competitors. According to one analyst, "[b]y offering the strongest protection, the United States has stimulated more creativity and new industries than anywhere else--and an annual thirty billion dollar intellectual property trade surplus."[18] A patent

---

     9. Robert Holleyman, *Copyright Protection for Computer Software: A Global Overview*, in INTELLECTUAL PROPERTY LAW 1995, 313 (PLI Patents, Copyrights, Trademarks, and Literary Porperty Course Handbook Series No. G4-3948, 1995).
     10. *Id.*
     11. Derek Leebaert, *News from the Frontiers*, in THE FUTURE OF SOFTWARE 1, 5 (Derek Leebaert ed., 1995).
     12. *Id.*
     13. Holleyman, *supra* note 9, at 313.
     14. Leebaert, *supra* note 11, at 5.
     15. Effy Oz, *Software Intellectual Property . . . Protection Alternatives*, J. OF SYSTEMS MGMT., 50, 56 (July 1995).
     16. Clair Whitmer, *Industry Divided Over Software Patents*, INFOWORLD, Feb. 28, 1994, at 20.
     17. David Vaskevitch, *Is Any of This Relevant?* in THE FUTURE OF SOFTWARE 45, 74 (Derek Leebaert ed., 1995).
     18. David Friedman, *A Policy that Punishes American Ingenuity*, L.A. TIMES, Nov. 19, 1995, at M2.

can give its owner the advantage of a long-term outlook, and it may help
parties to determine whether to fight over a market niche.[19] This effect helps
stabilize industry and minimizes disputes by making clear boundaries
between competitors.[20]

Patents can be used either offensively or defensively. Offensively, a
patent can be used to gain a limited monopoly for a product. This monopoly
can be used to block competition and establish the patent owner firmly in the
market. When used defensively, a patent can keep competitors from
acquiring broad patents that shut everyone else out of a particular area. As
software developer and entreprenuer Paul Heckel says:

> Patents, like a cat's claws, function as weapons when necessary.
> A declawed cat will not survive in the wild; neither can a
> defenseless startup once it succeeds and attracts substantial
> competitors. Patents are not the only defense, but they are vital
> to innovative startups that must survive. In business, as in the
> jungle, respect is given only to those who can protect
> themselves.[21]

Heckel feels that much of a patent's value is as a potential threat. [22] He
claims that software patents will help the industry by stimulating companies
to bring commercial products to the market, and by stimulating the formation
of new businesses.[23] An existing patent may force competitors to innovate
their way around the patent, thereby producing new and useful products that
do not infringe on the original patent. Heckel also believes that patents can
protect small innovators against big companies who would muscle them out
of the market without patent protection.[24]

C. *The European View*

The attitude toward software protection in the European Union (EU)
is different from the attitude in the United States. The EU States are barred
by treaty from granting patents on software,[25] and are consequently

---

19. Paul Heckel, *Debunking the Software Patent Myths*, 35 COMM. OF THE ACM, No.
6, at 132 (June 1992).

20. *Id.*

21. *Id.* at 129.

22. *Id.* at 134.

23. *Id.*

24. *Id.* at 135.

25. Convention on the Grant of European Patents (European Patent Convention), Oct.
5, 1973, 13 I.L.M. 270, art. 52(2)(c) [hereinafter EPC].

encouraged to rely on copyright for protection.[26] Under the European Patent Office (EPO), patent applications are published eighteen months after the patent application is filed.[27] The Europeans view this publication of the patent as risky.[28] They believe that the legal process is too protracted to be useful for protection against infringement in an area such as software, where there is rapid technological change.[29] The European Strategic Programme for Research and Development in Information Technology (ESPRIT) Report acknowledges that European copyright protection offers less protection "than that conferred by the law of some countries, and the difference operates to the detriment of those who hold European rights."[30] The ESPRIT report also states that, as a result, "undertakings tend to fall back on secrecy."[31] The EPO does not believe that any changes in patent law are necessary, as "[t]he approach initiated by the revised guidelines in 1985 has proved basically satisfactory and offered wide scope for patenting software technology."[32] The EPO does not totally rule out changes in its patent law, admitting that:

> [i]n the USA, a number of decisions of the court of Appeals for the Federal Circuit in Washington have made it much easier to patent any type of computer software, thereby further lowering the barriers against software patents. The EPO will be analysing these decisions carefully to see whether thay have any bearing on European patent practice.

It has recently been suggested that the solution to the problem of protecting computer software lies in discarding the section of Article 52 EPC which prohibits the patenting of computer programs as such. Those who advocate this approach cite the WTO/TRIPS agreement which calls for

---

26. "Software is not protected by universal regulations such as apply to patents. The community did make provision for its own system of protection, by means of the directive of 14 May 1991(34), which adapted the system of copyright which applies to literary and artistic works." Special Report No. 6/93 Concerning the European Research and Development Programmes in the Field of Information Technology (the Esprit Programmes) Together with the Commission's Replies 1994 O.J. (C 45) 1, § 2.33 *available in* WESTLAW, CELEX-LEG Database [hereinafter Special Report No. 6/93].

27. EPC, *supra* note 25, art. 93.

28. "[S]ince a decision concerning the possible grant of a European patent can rarely be made so quickly, an applicant for a European patent is in effect forced to publish the subject matter of his application without knowing whether his application will be successful . . . ." GERALD PATERSON, THE EUROPEAN PATENT SYSTEM 20 (1992).

29. Special Report No. 6/93, *supra* note 26, § 2.32.

30. *Id.* § 2.33.

31. *Id.*

32. Patenting Computer Software 1994 Annual Report, European Patent Office (copy on file in the IND. INT'L & COMP. L. REV. office).

unrestricted patent protection for computer programs. Even if the EPO does not share this view, the suggestions will have to be more carefully evaluated when the time comes to revise the EPC.[33]

While not completely closing the door on any changes, it is apparent that it will be a major undertaking to make software per se patentable in the European Union.

D. *Pros and Cons of Software Patents*

The issue of whether software should be patentable, like any such debate, has valid points both pro and con. There are several commonly given reasons as to why computer programs should not be patented. These reasons reflect the difficulty of granting a patent on something that cannot be seen or touched. The most common reason given is that a person cannot patent an algorithm or a mathematical formula. An algorithm is a series of sequential steps used to solve a problem. Software designers usually draw up an algorithm before starting to write a computer program. It is essentially a strategy for solving a problem using a computer. The resulting computer program is an implementation of that algorithm, just as the first atomic bomb was an implementation of the mathematical equation $E=mc^2$. Another reason given for not granting a patent is the problem of clearly specifying the components and operation of the computer program. Source code cannot be used to specify the invention,[34] and the specification may well determine if the claimed computer program is found to be statutory. Patent drafters must be very careful to include a thorough description of the process or machine in the specification. Another common reason given for denying software patents is the policy against taking laws of nature out of the public domain. Such "preemption" of a law of nature is not tolerated, as laws of nature must remain free for all to use.

There are several compelling reasons why patents should be granted on computer programs. A computer program, when reduced down to its essence, is a system of electronic controls which manipulate electronic signals as they travel through a computer. When a computer performs an action, the computer program turns transistors in the machine on or off in a predetermined sequence. The program converts a general configuration of computer hardware into a specific configuration by effectively "rewiring" it. The same effect can be produced by actually constructing an equivalent logic circuit with electronic components. It has been argued that if the physical

---

33. *Id.*
34. Request for Comments on Proposed Examination Guidelines for Computer-Implemented Inventions, 60 Fed. Reg. 28,778 I(B)(2)(a) (1995) [hereinafter PTO Guidelines] (proposed June 2, 1995).

circuit is patentable, why is the software that configures the machine unpatentable?[35] The PTO Guidelines would allow almost all software to be patented. Another argument in favor of patenting software is that computer software is not an algorithm, but rather it is an implementation of an algorithm. Much confusion and needless complication has arisen through courts trying to determine patentability based on whether the patent claims were attempting to patent a computer algorithm. This test has not helped to determine what is patentable. It only clouds the issue by trying to apply a characterization that is completely unhelpful. The term "mathematical equation" is similarly unhelpful. Many, if not most, inventions are based on mathematical equations. Electronic circuits certainly can be based on equations, yet electronic patents are not rejected on those grounds.

Another compelling reason to grant intellectual protection to software is due to its status as a large and growing industry. Computer software is a very important industry, and it continues to grow in importance as the electronics industry grows. Even if the current rate of growth levels off, there is no doubt that in the future, computers and their programs will be running much of people's lives. Economic reasons have not been previously cited as a reason why software should be patentable, but the economic reality cannot be ignored. The software industry needs the protection of intellectual property law. The need for this protection will grow as the industry grows. The move toward software patents seems inexorable, and arguably, "[a]ttempting to make software unpatentable will no more prevent practical software patents from issuing and being enforced than prohibition will eliminate alcoholism."[36]

---

35. "[A]ny implementation carried out by programming a computer can also be carried out in hardware, so that premising patentability on one of the two makes no scientific sense." James R. Goodman et al., *Toward a Fact-Based Standard for Determining Whether Programmed Computers are Patentable Subject Matter: the Scientific Wisdom of Alappat and Ignorance of Trovato*, 77 J. PAT. & TRADEMARK OFF. SOC'Y 353, 353-54 (1995).

> If *Trovato* can be understood to require a disclosure of computer architecture and hardware of the programmed computer, then the way to obtain a patent is to simply convert the software implementation into a hardware implementation of the programmed computer, obtain a patent on the hardware, and show infringement under the doctrine of equivalents. Converting from software to hardware is routine . . . . Thus, requiring a hardware disclosure would be a mindless increase in the cost for a patent application, particularly where the best mode of the invention is a software implementation.

Id. at 357. "[S]imple software programs can infringe almost pure hardware patents [which] suggests the difficulty of drawing a legal distinction between hardware and software." Heckel, *supra* note 19, at 125.

36. Heckel, *supra* note 19, at 128.

1. *The Position of the United States Government*

The extensive public debate over the patenting of computer programs has given rise to groups advocating both more and less protection. The most important viewpoint to consider is that of the PTO because it determines what patents are issued. The PTO has long held that software is not patentable. The PTO has issued three separate sets of guidelines on patentability. The first guideline, published in 1968, stated that computer programs, per se, were not patentable. However, programs could be patented as part of a process or apparatus, but only if they operated on a physical quantity and did not simply manipulate abstract quantities.[37] The second guideline, issued in 1981, stated that if the patent claim as a whole merely recited an algorithm, then that claim did not recite statutory subject matter.[38] The second guideline also reaffirmed the statement made by the Supreme Court in *Diamond v. Diehr*[39] that a claim on a mathematical formula may be found to be non-statutory even if the claim is limited or does not totally preempt the formula.[40] The third guideline is the subject of this note and will be discussed later.

In the absence of broad patent protection for software, the applicability of copyright protection was investigated. In 1974, the United States Congress established the Commission on New Technological Uses of Copyrighted Works (CONTU). The Commission was created to determine whether copyright law could be used to adequately protect software without the need for software patents.[41] The CONTU report concluded that computer programs were an expression of ideas and therefore could be protected under copyright law.[42]

2. *The Position of the Software Industry*

The software industry is divided over the issue of patent protection. This struggle is caused by a tension between the distrust of monopolies and the belief that a person is entitled to his or her own creation and can therefore control and monopolize it.[43] Several groups are opposed to software patents. Some of the industry members that are against patenting

---

37. Examination of Patent Applications on Computer Programs, 68 FR 15609 (1968).
38. 1 DONALD S. CHISUM, PATENTS, A TREATISE ON THE LAW OF PATENTABILITY, VALIDITY AND INFRINGEMENT 103[6][h] (1995).
39. Diamond v. Diehr, 450 U.S. 175 (1981).
40. *Id.*
41. JEFFREY P. CUNARD, *Property of the Mind: Software and the Law, supra* note 11, 227, 232.
42. *Id.*
43. Oz, *supra* note 15, at 50.

software are Adobe Systems, Oracle Corp., Autodesk, Inc., and the 600 member League for Programming Freedom.[44] Both Lotus Development Corporation founder Mitch Kapor and the League for Programming Freedom feel that patents are being granted on fundamental technology, and such patents will hurt United States' competitiveness.[45] These opponents feel that software patents will stifle innovation and prevent all but the largest corporations from thriving.[46] The League relies on a survey taken at the 1991 SIGGRAPH conference to validate their position. The League claims that the survey shows industry support for patents on computer programs is low and getting lower.[47] One of the League's concerns is that an inventor who independently develops an idea at the same time as the patentor could be precluded from obtaining a patent.[48] The League's second concern is the PTO's "ignorance of the prior art."[49] If the PTO does not have an extensive catalogue of prior art, it will not be able to screen out new inventions from the prior art.[50] But the League's most powerful argument is the success of the United States software industry, which was achieved without the benefit of patent protection.[51]

Many industry members, including IBM, Microsoft, Apple Computers, Taligent, AT&T, and Intel, support patents on computer software.[52] These companies fear that foreign competition will capture the United States software market if they are not given patent protection.[53] The Association of Data Processing Service Organizations (ADAPSO) argues that because of the economic risk in the software industry software patents are needed as an

---

44. Whitmer, *supra* note 16, at 20.

45. CUNARD, *supra* note 41, at 252.

46. Whitmer, *supra* note 16, at 20.

47. Pamela Samuelson et al., *Developments on the Intellectual Property Front*, 35 COMM. OF THE ACM, No. 6, at 33, 34 (June 1992).

48. *Id.* at 35.

49. *Id.* at 36. Prior art has been defined as "knowledge in certain statutorily defined categories which predated the invention or, in some instances, which predated the application for patent by more than one year even though it may have been subsequent to the invention." ARNOLD & VADEN, *supra* note 2, at 28. "To obtain a patent, the applicant has to show that there is no 'prior art,' publications describing the idea or existing devices that are based on the idea. In the language of the law, the existence of such prior art renders the 'invention' obvious." Oz, *supra* note 15, at 55.

50. The claim is that because the PTO has not previously accepted computer software claims, it does not have the requisite database of "prior art" to use when examining patent applications. Pamela Samuelson, *Benson Revisited: The Case Against Patent Protection for Algorithms and Other Computer Program-Related Inventions*, 39 EMORY L.J. 1025, 1138-39 (1990).

51. Samuelson, *supra* note 47, at 38.

52. Whitmer, *supra* note 16, at 20.

53. *Id.*

incentive for new software inventions.[54] ADAPSO claims that software
patents may help in obtaining loans and attracting investors.[55] Software
patents may also be instrumental in attracting talent.[56] Another strong
argument is that software is no different than any other technology and
should not be treated differently.[57] In response to the argument that the
software industry has thrived without patents, software entrepreneur Paul
Heckel states "[w]hen there is no established competition, new companies
can compete without patents. As the industry matures it becomes difficult
for new companies to enter the market without a sustainable advantage such
as patents."[58] In the past computer programs may not have needed patent
protection in order for the industry to survive. Nevertheless, the software
industry would be well advised to play it safe and assume that innovation
occurs in the software industry just like it does in other industries.[59] As
worldwide industry gears up to gain more of the global software market,
United States industry will need the protection of the Patent Office to protect
the software industry and avoid the fate of the semiconductor industry.

E. *The Importance of Software Patents*

In reality, software patents already exist.[60] In many cases, software
developers can get patents on computer programs if they know the rules.
But the patent protection could be broader and easier to obtain. As some
have argued, patents on computer programs have to be obtuse to be granted.
Pamela Samuelson, an attorney for the League for Programming Freedom,
asserts that patent lawyers write patent applications in a less than
straightforward manner.[61] All patents should be as straightforward and
informative as possible. Obtuse patents will defeat the main goal of the
patent system, which is to teach the patent to the public upon its disclosure.
Straightforward software patents will be easier to obtain and more effective.

---

54. Brief for ADAPSO as Amicus Curiae at 44, Parker v. Flook, 437 U.S. 584 (1978)
(No. 77-642), *quoted in* Diamond v. Diehr, 450 U.S. 175, 218 n.42 (1981) (Stevens, J.,
dissenting).

55. *Id.*

56. *Id.*

57. Oz, *supra* note 15, at 56.

58. Heckel, *supra* note 19, at 126.

59. *Id.* at 137.

60. "Software, or 'computer-related' patents were obtained in the 1960s. Martin Goetz
of Applied Data Research received U.S. patents 3,380,029 in 1968 on a Sorting System, and
3,533,086 in 1970 on AutoFlow, an automatic flow charting program." *Id.* at 131.

61. *Id.* at 125.

II. HISTORY AND DEVELOPMENT OF SOFTWARE PATENT LAW

A. *Statutory Patent Law*

The United States Constitution established the patent system by granting Congress the power to "promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."[62] Pursuant to the Constitution, Congress enacted the Patent Act of 1793. The Act protects "any new and useful art, machine, manufacture, or composition of matter, or any new or useful improvement [thereof]."[63] The Supreme Court has held that by "choosing such expansive terms as 'manufacture' and 'composition of matter,' modified by the comprehensive 'any,' Congress contemplated that the patent laws should be given wide scope, and the relevant legislative history also supports a broad construction."[64] This definition of patentable subject matter was maintained through the Patent Acts in 1836, 1870, and 1874.[65] The Patent Act of 1952 codified then existing patent law.[66] The only change made by Congress was the replacement of the word "art" with the word "process."[67] Title 35 of the United States Code contains the requirements for obtaining a patent. Section 101 of Title 35 governs the patentability of subject matter. "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, subject to the conditions and requirements of this title."[68]

B. *Development of United States Case Law*

The earliest computer software case was *In re Prater.*[69] In the patent, Prater claimed a method of spectrographic analysis useful in determining the identity of an unknown gas or gases. The claim was for an improved method of solving one particular subset of equations. This new method gave a more accurate determination of the makeup of the gas over the prior art. The United States Court of Customs and Patent Appeals held that Prater's process

---

62. U.S. CONST. art. I, § 8, cl. 8.

63. Diamond v. Chakrabarty, 447 U.S. 303, 308 (1980) (quoting Patent Act of Feb. 21, 1793, § 1, 1 Stat. 319).

64. *Id.* at 303.

65. *Id.* at 309.

66. S. Rep. No. 1979, 82d Cong., 2d Sess. 5 (1952); H.R. Rep. No. 1923, 82d Cong., 2d Sess. 6 (1952).

67. *Id.*

68. 35 U.S.C. § 101 (1994).

69. *In re* Prater, 56 C.C.P.A. 1376 (1968).

and apparatus claims were patentable subject matter under 35 U.S.C. § 101.[70] The court held that claims for a process to be performed on a computer were not invalid just because they could be performed in a person's mind.[71]

The patent application in *Gottschalk v. Benson* claimed a method for converting binary-coded decimal numerals into pure binary numerals for use by a computer.[72] The Supreme Court held the patent to be invalid, stating that ideas or mathematical formulas cannot be patented.[73] The Court held that Benson's claims were broad enough that the patent would preempt any application of the mathematical formula contained in the patent application.[74] However, the Court did not find software per se to be unpatentable. The court stated that computer programs might be patentable as long as they met prior precedents.[75]

The next software case was *Parker v. Flook.*[76] Flook's patent application claimed a method of updating alarm limits in a catalytic conversion process. In *Flook*, the United States Supreme Court held that the use of a computer as part of a method of monitoring alarm limits was non-statutory.[77] The Court held that the mathematical formula implemented in the computer software was the only novel feature.[78] *Flook* held that the adjustment of the alarm limit to the number calculated by the computer was just "post-solution activity," and as such it was not enough to make the idea patentable.[79] One part of the holding went so far as to claim that "[t]he process itself, not merely the mathematical algorithm, must be new and useful. Indeed, the novelty of the mathematical algorithm is not a determining factor at all."[80] This statement, implying that one novel element was not enough for patentability, was not supported by later holdings.

In 1981, the Supreme Court decided the landmark case of *Diamond v. Diehr.*[81] In the patent the co-inventors, Diehr and Lutton, claimed an improved process for molding and curing synthetic rubber. The rubber mold had to be opened at a certain time and temperature to ensure a quality

---

70. *Id.*

71. *Id.* at 1374. This was the so-called "mental steps" doctrine.

72. Gottschalk v. Benson, 409 U.S. 63, 71 (1972).

73. *Id.*

74. *Id.*

75. "We do not hold that no process patent could ever qualify if it did not meet the requirements of our prior precedents. It is said that the decision precludes a patent for any program servicing a computer. We do not so hold." *Id.*

76. Parker v. Flook, 437 U.S. 584 (1978).

77. *Id.*

78. *Id.* at 588.

79. *Id.* at 590.

80. *Id.* at 591.

81. Diamond v. Diehr, 450 U.S. 175 (1981).

molding.[82] The inventors incorporated a computer into the process to closely monitor mold temperatures and the appropriate time to open the molds. The computer used Arrhenius' equation[83] to calculate the proper opening time. Even though the computer used by Diehr and Lutton implemented a known mathematical equation, the formula did not render the patent invalid.[84] If a computer is part of a patentable process, the process is still patentable even if the mathematical formula (or computer algorithm) is not patentable on its own.[85] The Supreme Court held that Diehr's patent claims were patentable subject matter. The Court decided that the application of Arrhenius' equation did not preempt the formula since its use was narrowly limited to this particular process.[86] In the case of *Benson*, it is apparent that Benson's patent, if granted, would have taken a law of nature out of the public domain. However, it is not always that easy to tell if a patent would preempt a law of nature or a mathematical formula. Although Diehr's patent for curing synthetic rubber might have been viewed as removing a law of nature from the public domain, it was granted by the Patent Office. The difference between the two is in the breadth of the claims. Benson claimed a broad method of converting a binary-coded decimal number into a pure binary number. Diehr's patent claim specifically limited the application of a mathematical formula to determine the correct time to open a rubber curing mold. In *Diehr*, the Court instructed the Patent Office to determine statutory subject matter by looking at all claims as a whole and not the novelty of the claim.[87] The Court reiterated *Benson* by stating that the legislature had not placed any limitations on computer software patents in Title 35 of the United States Code.[88] It is interesting to note that the process in *Diehr* is very similar to the process in *Flook*.[89] What had been previously unpatentable in *Flook* was now considered statutory subject matter.

The dissent in *Diehr* asserted that Diehr and Lutton's invention was not unique. It was novel only in that the opening time of the rubber mold was calculated by a computer instead of a human.[90] The dissent argued that Diehr was attempting to patent an algorithm, and therefore the claim was

---

82. *Id.* at 177.

83. Arrhenius' equation was named after its discoverer Svante Arrhenius. It is of the form $\ln v = CZ + x$, where v is the total required cure time, C is the activation constant, Z is the temperature in the mold, and x is a constant dependent on the geometry of the mold. *Id.* at 178 n.2.

84. *Id.* at 188-89.

85. *Id.* at 192.

86. *Id.* at 187.

87. *Id.* at 188-89.

88. *Id.* at 187.

89. Both monitored a chemical process with a computer to determine when the process was done.

90. *Diehr*, 450 U.S. at 207-08.

unpatentable under 35 U.S.C. § 101.[91]   The computer application of
Arrhenius' equation was just the "discovery" and application of a law of
nature and mathematical principal.[92] The dissent raised two valid concerns:
precedent did not provide any hard rules for inventors, and the term
"computer algorithm" was too vague and over-inclusive.[93]

In *Arrhythmia Research Technology, Inc. v. Corazonix Corporation*,
the patent application claimed a method and apparatus for analyzing
electrocardiographic signals as a way of monitoring the heart activity of
heart-attack victims.[94] The Court of Appeals for the Federal Circuit (CAFC)
held that the patent was valid. The court held that the inputs and outputs to
the computer were not abstract numbers but were, however, related to the
victim's heart activity.[95] The court further held that the process was statutory
because the "claimed steps of 'converting,' 'applying,' 'determining,' and
'comparing' are physical process steps that transform one physical, electrical
signal into another."[96] The court declared that "[t]he claimed invention . .
. converts one physical thing into another physical thing just as any other
electrical circuitry would do."[97]

The next landmark case in computer software patents was *In re
Alappat*.[98] Alappat and his co-inventors claimed a means for smoothing a
waveform display on a digital oscilloscope.[99] The CAFC reiterated the
Supreme Court's statement in *Benson* that there were no limitations on
computer software patents in 35 U.S.C. § 101 or any legislative history.[100]

---

91. *Id.* at 212-15.

92. *Id.* at 194, 207-08.

93. *Id.* at 219.

94. Arrhythmia Research Technology, Inc. v. Corazonix Corporation, 958 F.2d 1053
(Fed. Cir. 1992).

95. *Id.* at 1059.

96. *Id.*

97. *Id.* at 1060 (quoting *In re* Sherwood, 613 F.2d 809, 819 (C.C.P.A. 1980), *cert.
denied*, 450 U.S. 994 (1981)).

98. *In re* Alappat, 33 F.3d 1526 (Fed. Cir. 1994).

99. *Id.* A digital oscilloscope is an instrument used to display electronic waveforms.
A digital oscilloscope display screen has a fixed number of dots of light (pixels) it uses to
display waveforms, similar to a TV picture tube. When the waveform changes abruptly, the
displayed waveform can appear discontinuous or jagged. Alappat's invention applies an anti-
aliasing system to smooth out such irregularities in the displayed waveform. *Id.* at 1537.

100. *Id.* at 1542.

> The plain and unambiguous meaning of s 101 is that any new and useful
> process, machine, manufacture, or composition of matter, or any new and
> useful improvement thereof, may be patented if it meets the requirements for
> patentability set forth in Title 35, such as those found in ss 102, 103, and 112.
> The use of the expansive term 'any' in s 101 represents Congress's intent not
> to place any restrictions on the subject matter for which a patent may be
> obtained beyond those specifically recited in s 101 and the other parts of Title
> 35. Indeed, the Supreme Court has acknowledged that Congress intended s 101

The CAFC acknowledged that the laws of nature, natural phenomena, and abstract ideas are excluded from patentability.[101]   However, the court explained that a claim containing a mathematical formula and an implementation of that formula in a machine or process, when "considered as a whole, is performing a function which the patent laws were designed to protect . . . then the claim satisfies the requirements of § 101."[102]   The CAFC stated that a computer program creates a new machine when it programs the computer.[103] The *Alappat* court did not stop there. It took the bold and prescient step of declaring that a computer program is statutory subject matter if it meets all other requirements of patentability.[104] This was the first real acknowledgment that software is patentable on its own.

The dissenting judges in *Alappat* felt Alappat was trying to patent a mathematical discovery and not an invention.[105] The dissent stressed that Congress is directed by the Constitution to promote the "useful arts" and not "science."[106] Patents are supposed to be rewards to entice people to invent technologically useful applications of science and not just theoretical discoveries.[107] The dissent argued that the crucial issue was if an invention was determined to be more than just mathematics, it had to be a technologically useful application of mathematics.[108]

In *In re Warmerdam*, the patent application claimed a method and apparatus for preventing collisions between computer-controlled machines and their surroundings by using data structures called bubble hierarchies.[109] The CAFC held that the claims regarding the use of bubble hierarchies for preventing collisions were not patentable because they were merely manipulations of abstract ideas.[110] The court relied on *Alappat*, saying that claims must go beyond simply manipulating abstract ideas or natural phenomena.[111] However, the court held that the claim for a machine with a memory containing data representing a bubble hierarchy was patentable.[112]

---

to extend to 'anything under the sun that is made by man.'
Id. at 1542 (quoting Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980)).
    101. *Id*. at 1543.
    102. *Id*.
    103. *Id*. at 1545.
    104. *Id*. The requirements of Title 35 referred to by the CAFC are found in 35 U.S.C. § 102, 103, and 112.
    105. *Id*. at 1552-54.
    106. *Id*. at 1552.
    107. *Id*.
    108. *Id*. at 1557.
    109. *In re* Warmerdam, 33 F.3d 1354 (Fed. Cir. 1994). Bubble hierarchies are defined as artificial circular boundaries that are used to detect potential collisions between objects. If a potential collision is detected, the boundary is reduced to a smaller bubble zone. *Id*. at 1355.
    110. *Id*. at 1359-60.
    111. *Id*. at 1360.
    112. *Id*. at 1360-61.

In *In re Trovato*, the patent application claimed a method for calculating the optimal path between two locations by solving the shortest path between points.[113] The CAFC held that Trovato's patent was non-statutory. The patent application did not disclose any apparatus. The entire disclosure of the patent consisted of flow charts and program code.[114] The CAFC felt that Trovato was trying to indirectly patent a mathematical algorithm by wording the claim as a machine.[115] A major difficulty for the court was that Trovato's invention did not manipulate any physical quantities.[116] Even though *Alappat* had already been decided, the CAFC seemed reluctant to accept software as patentable subject matter. Trovato's case was weakened by the fact that the patent application did not describe how the calculated values were used or how the computer program would configure and use the computer hardware.[117] To make matters worse, the CAFC typified Trovato's inclusion of an electronic readout of the computer data as "mere post-solution display."[118] The applicant's petition for a rehearing was granted on July 25, 1995. The CAFC granted the petition in light of *Alappat*[119] and the PTO's Proposed Guidelines.[120] The dissent argued that there was no basis for a rehearing because the law had not changed.[121] The PTO Guidelines were issued to garner feedback from the software industry and the legal community, thus it does not have any legal force on the court. The dissent also argued that the patent application did not disclose any specific structure for its claims.[122]

---

113. *In re* Trovato, 42 F.3d 1376 (Fed. Cir. 1994) *vacated*, 60 F.3d 807 (Fed. Cir. 1995).

114. *Id.* at 1380.

115. *Id. at* 1382-83.

116. *Id.* at 1381.

117.   Trovato's applications fail even to explain how the claimed inventions actually employ the numbers derived to control movement. Although the inventions likely employ techniques known to the art to move an object along the lowest cost path it calculates, the absence of even a cursory description of how the computed values are implemented further indicates that the claimed methods comprise only numerical manipulation.

Id. "Indeed, the specifications note the inventions' 'general applicability to numerical methods' and seek to describe them '[f]rom a mathematical point of view.'" *Id.* at 1380.

118. *Trovato*, 42 F.3d at 1380.

119. *Trovato* was originally heard by the PTO Board of Appeals on July 22, 1992, and May 26, 1993, Appeal Numbers 92-1843 and 92-4106 respectively. The CAFC decided *Alappat* on July 29, 1994. The CAFC originally decided *Trovato* on December 19, 1994. The PTO issued guidelines for software patents on June 2, 1995.

120. *In re* Trovato, 60 F.3d 807 (Fed. Cir. 1995).

121. *Id.*

122. *Id.*

In *Etak v. Zexel*, Etak's patent claimed a computer navigation apparatus for use in a vehicle.[123] The U.S. District Court for the Northern District of California held that Etak's patent was statutory subject matter. The court reaffirmed the statement that an application of an algorithm is not barred by 35 U.S.C. § 101.[124] The court listed three types of claims it considered to be non-statutory: a new method of producing a number, antecedent data gathering steps, and post-solution displays of computer generated results.[125] The court also reiterated the idea first presented in *Alappat* that a computer program makes a general purpose computer into a new and unique machine.[126] The court distinguished this case from *Trovato* in that Etak's specification recited some hardware while Trovato's did not.[127] The court reaffirmed the holding of *Diehr*, stating that a combination of known elements (hardware and software) could be patentable even if the individual components were not patentable.[128]

In *In re Beauregard*, the Commissioner of Patents and Trademarks moved to dismiss the petitioner's appeal.[129] The CAFC vacated the Board of Patent Appeals and Interferences' rejection of the patent application in light of the Commissioner's statement that "computer programs embodied in a tangible medium, such as floppy diskettes, are patentable subject matter under 35 U.S.C. § 101."[130] The Commissioner of Patents' statement was a clear signal that the PTO was changing its position on software patents.

The Supreme Court has not ruled on the patentability of a computer program since 1981 when it decided *Diehr*. The CAFC has decided all of the recent cases except *Etak*. It is apparent from the cases above that the positions of the PTO and the courts are evolving. At the start, both viewed computers and computer programs as little more that abstract mathematical formulas. The current position is slowly gaining acceptance. If any trend can be extrapolated from the case law, it is a trend of increased protection as software grows in importance in the American economy.

---

123. Etak, Inc. v. Zexel USA Corp., No. C 94-4041, 1995 WL 462240 (N.D. Cal. May 8, 1995).
124. *Id.* at 2.
125. *Id.*
126. *Id.* at 5.
127. *Id.*
128. *Id.*
129. *In re* Beauregard, 53 F.3d 1583 (Fed. Cir. 1995).
130. *Id.*

III. DISCUSSION OF CURRENT EUROPEAN UNION LAW

A. *Convention on the Grant of European Patents*

The Convention on the Grant of European Patents, known as the European Patent Convention (EPC), was created for the purpose of eliminating the patent boundaries between member states as part of the effort to form the European Economic Community (EEC).[131] The European Union was formed on November 1, 1993, when the Treaty on European Union[132] (Maastricht Treaty) went into effect. As of 1992 there were twelve member states in the EU: Belgium, Denmark, France, Germany, Greece, Ireland, Italy, Luxembourg, the Netherlands, Portugal, Spain, and the United Kingdom.[133] The EPC came into force in 1978,[134] and all EU states have signed the EPC.[135] Effectively the EPC is one application and one examination process. After the application is granted, applicants receive national patents in all designated countries.[136] However, the European Patent Office (EPO)[137] does not replace the national patent offices of the member states.[138] The EU members did not make the EPC the sole patent provider because they realized that the individual states would not immediately change over from their national patent systems to the EU system.[139] Inventors can still simply file a national patent if they do not want or need a EU patent.[140] When an EPC patent application is filed, the EPO determines patentability of the subject matter. If the application is filed only in a member state, that state's patent law determines patentability.[141] Unlike the United States Code and court decisions, the EPC unequivocally excludes patents on computer

---

131. PATERSON, *supra* note 28, at 2.

132. Treaty on European Union (Maastricht Treaty), February 1, 1992, 31 I.L.M. 247 (entered into force November 1, 1993).

133. PATERSON, *supra* note 28, at map preceding 1.

134. *Id.* at 2.

135. *Id.* at map preceding 1.

136. *Id.* at 20.

137. The European Patent Office was established by the EPC as part of the European Patent Organisation. The EPO was given the responsibility of granting patents under the treaty. EPC, *supra* note 25, art. 4. The Organisation and Office were established in Munich, Germany. *Id.* art. 6.

138. Jeffrey L. Thompson, Note, *The North American Patent Office? A Comparative Look at the NAFTA, the European Community, and the Community Patent Convention*, 27 GEO. WASH. J. INT'L L. & ECON. 501, 510 (1993). *See also* EPC, *supra* note 25, arts. 2 and 66.

139. PATERSON, *supra* note 28, at 19-20.

140. Thompson, *supra* note 138, at 510.

141. "The coexistence of the European patent system with existing national systems has been generally recognized as the main reason for its international acceptability." PATERSON, *supra* note 28, at 20.

programs from its categorization of patentable subject matter.[142]   This
exclusion is qualified by the statement that the excluded subject matter is
barred from patentability "only to the extent to which a European patent
application or European patent relates to such subject-matter or activities as
such."[143]   In other words, a computer program cannot be patented on its
own. The rationale behind this exclusion is that a computer program is not
an "invention" because it is essentially abstract in character.[144]

B. *Community Patent Convention*

The Community Patent Convention (CPC)[145] was initiated shortly after
the EEC Treaty became effective in 1958.[146] The goal of the CPC was to
further open borders to trade within the European Community (EC) by
eliminating differences in patent enforcement between the member states.
Nine countries initiated and signed the first draft of the CPC in 1975.[147]
Article 98 provided that the convention would not go into effect until it was
ratified by all of the EC countries.[148] It has yet to be ratified by all EU

---

142. EPC art. 52 reads in part:
   (1) European patents shall be granted for any inventions which are susceptible
   of industrial application, which are new and which involve an inventive step.
   (2) The following in particular shall not be regarded as inventions within the
   meaning of paragraph 1:
   (a) discoveries, scientific theories and mathematical methods;
   (c) schemes, rules and methods for performing mental acts, playing games or
   doing business, and programs for computers.
EPC, *supra* note 25, art. 52(1), (2).
   143. EPC, *supra* note 25, art. 52(3).
   144. PATERSON, *supra* note 28, at 314.
   Whatever their differences, these exclusions have in common that they refer to
   activities which do not aim at any direct technical result but are rather of an
   abstract and intellectual character. The requirement that an invention must have
   a technical character, or in other words, must provide a technical contribution
   to the art, is at the basis of a long-standing legal practice in at least the majority
   of Contracting States of the EPO. Neither from the terms of Article 52 EPC,
   nor from the legislative history of that Article as appearing from the preparatory
   documents, can it be deduced that these Contracting States would have intended
   to deviate from their national laws and jurisprudence in this respect. On the
   contrary, it seems to be borne out by the list of exceptions in Article 52(2)(a)
   to (d) EPC that they did not wish to do so.
Id. at 314-15 (quoting T22/85 (IBM/Document abstracting and retrieving) O.J. EPO 1990,
12).
   145. Convention for the European Patent for the Common Market (Community Patent
Convention), 1975 O.J. (L 017) 1 [hereinafter CPC], *available in* WESTLAW, CELEX-LEG
Database.
   146. Thompson, *supra* note 138, at 511.
   147. PATERSON, *supra* note 28, at 21.
   148. Thompson, *supra* note 138, at 512 n.90.

member states.[149] The CPC will convert the bundle of patents of the EPC
into one unified patent for all of the member states.[150]   This single
Community Patent will be enforced in all member states with a Common
Appeal Court given exclusive jurisdiction to determine issues raised on
appeal and a single body of law within the Convention itself.[151] The CPC
maintains the EPC's prohibition against computer program patents. [152]

C. *National Laws of the European Community Member States*

The member states of the EU generally have the same prohibition
against the patenting of computer programs as the EPC.[153] The EU members
that specifically prohibit the patenting of computer programs are:
Belgium,[154] Denmark,[155] France,[156] Germany,[157] Greece,[158] Ireland,[159] Italy,[160]

---

149. *Id.* at 511. The EC was supplanted by the EU by the Maastricht Treaty, Feb. 7,
1992. *See supra* note 132.

150. PATERSON, *supra* note 28, at 22.

151. Thompson, *supra* note 138, at 512.

152. Article 56 of the CPC states that any person can contest a patent by filing an
application for revocation. CPC, *supra* note 25, art. 56(1), 475A3490 at 19. The applicable
grounds for filing an application for revocation with respect to a computer program would be
that the subject matter is excluded from patentability under Articles 52 to 57 of the EPC.
CPC, *supra* note 25, art. 57(1)(a), at 17-20.

153. *See supra* note 144 for the EPC's provision excluding software from patentability.

154. New Belgian Patent Law of March 28, 1984, ch. II, pt. 1, § 3(1)(3), *translated in*
2C JOHN P. SINNOTT & WILLIAM J. COTREAU, WORLD PATENT LAW AND PRACTICE,
Belgium-12 (1995).

155. Patents Act. No. 479 of December 20, 1967, amended June 7, 1989 pt. 1, § I(2)(iii),
translated in 2C JOHN P. SINNOTT & WILLIAM J. COTREAU, WORLD PATENT LAW AND
PRACTICE, Denmark-43.

156. French Patent Law, ch. 1, art. 6(2)(c), *translated in* 2D JOHN P. SINNOTT, WORLD
PATENT LAW AND PRACTICE, France-111 (1995).

157. Patent Law, pt. I, § 1(2)3. *translated in* 2D JOHN P. SINNOTT, WORLD PATENT LAW
AND PRACTICE, West Germany-78.21.

158. Law 1733/87, pt. II, Chap. 1, art. 5(2)(c), *translated in* 2E JOHN P. SINNOTT,
WORLD PATENT LAW AND PRACTICE, Greece-10 (1994).

159. Patents Act, 1992, pt. II, ch. II, 9(2)(c), *reprinted in* 2E JOHN P. SINNOTT, WORLD
PATENT LAW AND PRACTICE, Ireland-88 (1994).

160. Italian Royal Decree of 29th June 1939, *as amended by* June 22, 1979, tit. II, ch.
I, § 12(b), *translated in* 2F JOHN P. SINNOTT & WILLIAM J. COTREAU, WORLD PATENT LAW
AND PRACTICE, Italy-5 (1995).

Spain,[161] the Netherlands,[162] and the United Kingdom.[163]   Portugal [164] and Luxembourg[165] do not have statutory provisions prohibiting computer program patents.

D. *European Cases on the Patenting of Computer Programs*

The leading European case on computer program patents is *Vicom*.[166] The patent applicant, Vicom Systems Inc., claimed an invention for the digital processing of images. The images were input in the form of a data array.   The EPO Technical Board of Appeal[167] held that the patent application was valid. The issue before the Board was whether the patent must be excluded under EPC 52(2) and (3) because it related to a computer program as such.[168] The Board held that the computer program was part of a technical process, and a computer controlled technical process is not statutorily barred by EPC 52(3).[169] Since the patent application did not attempt to patent just a computer program, it did not violate the exclusion "as such." The Board declared that even though computer programs under the EPC are non-statutory subject matter, including a computer program in the

---

161. Royal Decree 2424/1986, of 10 October, Relating to the Application of the Convention on the Grant of European Patents Made at Munich on 5 October, 1973, pt. I, art. 4, § 2(c) *translated in* 2H JOHN P. SINNOTT, WORLD PATENT LAW AND PRACTICE, Spain-94 (1994).

162. The Patents Act (Rijksoctrooiwet) *as amended by* May 29, 1987 *translated in* 2 LESTER NELSON, DIGEST OF INTELLECTUAL PROPERTY LAWS OF THE WORLD, Netherlands-2 (1995).

163. Patents Act 1977, § 1(2)(c), *reprinted in* GERALD PATERSON, THE EUROPEAN PATENT SYSTEM 785 (1992).

164. Portugese Industrial Property Act, Decree No. 30,679 of August 24, 1940, *as amended by* 1987, *translated in* 2H JOHN P. SINNOTT, WORLD PATENT LAW AND PRACTICE, Portugal-3 (1995).

165. The Law on Patents of Invention of June 30, 1880 *as amended by* April 27, 1922, *translated in* 2 LESTER NELSON, DIGEST OF INTELLECTUAL PROPERTY LAWS OF THE WORLD, Luxembourg-2 (1995).

166. T208/84 (VICOM/Computer-related invention) O.J. EPO 1987, 14, *reprinted in* 2M JOHN P. SINNOTT & WILLIAM J. COTREAU, WORLD PATENT LAW AND PRACTICE EPD-622 (1995).

167. The Board of Appeals hears appeals from persons contesting the decisions of the Receiving Section, Examining Divisions, Opposition Divisions, or Legal Division. EPC, *supra* note 25, art. 21.

168. 2M SINNOTT & COTREAU, *supra* note 166, at EPD-622.

169. The Board declared that a mathematical method used in a process does not render that process non-statutory if it is "used in a technical process, [and] that process is carried out on a physical entity (which may be a material object but equally an image stored as an electric signal) by some technical means implementing the method and provides as its result a certain change in that entity." *Id.* at EPD-626.

invention will not make the invention unpatentable as well.[170] The court
stated that the digital filtering/signal processing involved in the patent was
a "real world" activity that went beyond the abstract mathematical methods
prohibited by the EPC.[171] A computer process that changes some physical
entity, not just numbers, is patentable.[172] The Board held that the electrical
signals which represented the digital picture were the physical entities
changed by the process. It is remarkable that the Board took this view, in
light of the fact that the data input and output by the "process" was an array
of numbers. In many ways the Board's holding resembled the United States
decision in *Diamond v. Diehr*,[173] and even more closely paralleled the
holding in *In re Alappat*.[174] The most interesting statement made by the
Technical Board of Appeal was that "it would seem illogical to grant
protection for a technical process controlled by a suitably programmed
computer but not for the computer itself when set up to execute the
control."[175]

In the case of *Merrill Lynch's Application*,[176] the patent application
claimed an automated securities trading system which analyzed and
processed customers' buy and sell orders. *Merrill Lynch* claimed the
invention as a process. This patent application was made under United
Kingdom national law and the case was heard by the United Kingdom
Patents Court, not the EPO Board of Appeal. The United Kingdom Patents
Court held that the application was invalid. The Patents Court reasoned that
if the only inventive step resided in a computer then the invention as a whole
was not patentable.[177] Since the idea underlying the invention was to use a
computer to perform a known function that had previously been performed
by humans, the Patent Court held that what Merrill Lynch was trying to
patent was a computer program as such. The Court even went so far as to
hint that section 1(2)(c) of the Patents Act of 1977[178] contemplates a patent

---

170. *Id.* at EPD-627.
171. *Id.* at EPD-624.
172. *Id.* at EPD-626.
173. In *Diehr*, the U.S. Supreme Court held that even though a computer program
standing alone was not statutory subject matter, it was patentable as part of a process.
Diamond v. Diehr, 450 U.S. 175 (1981).
174. In *In re Alappat*, the CAFC held that a computer program used to smooth waveforms
displayed by a digital oscilloscope was a patentable machine. *In re* Alappat, 33 F.3d 1526
(1994). It is interesting that the claimed machine in *Alappat* was remarkably similar to the
claimed process here.
175. 2M SINNOTT & COTREAU, *supra* note 166, at EPD-6627.
176. [1988] R.P.C. 1. *cited in* PATERSON, *supra* note 28, at 316.
177. 2M SINNOTT & COTREAU, *supra* note 167, at EPD-823.
178. United Kingdom Patents Act of 1977 states in part:
    1.—(1) A patent may be granted only for an invention in respect of which the
    following conditions are satisfied, that is to say—
        (a) the invention is new;

to be non-statutory if it involves one of the excluded subjects, even if that subject is not the only inventive idea in the patent claim.[179] The court's logic was that if the invention was useful but was no more than a computer program, the invention could not be patentable.[180] The court was influenced by the fact that the claims were for a process which operated only on numbers and on its face was no more than a "mere method of doing business."[181] The *Merrill Lynch* court concurred with the EPO Board of Appeal's definition of a process as "providing a resulting change in the physical entity on which the process is carried out."[182] Merrill Lynch's patent application claimed no such change in a physical entity.

In the case of *Koch and Sterzel*,[183] the patent application claimed a computer-controlled X-ray apparatus. The Board of Appeal held that Koch's patent was statutory. The reasoning of the Board was that the invention had to be assessed as a whole, and even though a computer program as such was non-statutory, a mix of "technical" and "non-technical" features was patentable.[184] The Board stated that "[t]he EPC does not ask that a patentable invention be exclusively or largely of a technical nature; in other

---

(b) it involves an inventive step;
(c) it is capable of industrial application;
(d) the grant of a patent for it is not excluded by subsections (2) and (3) below;
and references in this Act to a patentable invention shall be construed accordingly.
(2) It is hereby declared that the following (among other things) are not inventions for the purposes of this Act, that is to say, anything which consists of—
(a) a discovery, scientific theory or mathematical method;
(b) a literary, dramatic, musical or artistic work or any other aesthetic creation whatsoever;
(c) a scheme, rule or method for performing a mental act, playing a game or doing business, or a program for a computer;
(d) the presentation of information.
Patents Act 1977 (U.K.) *reprinted in* GERALD PATERSON, THE EUROPEAN PATENT SYSTEM 785-86 (1992). Section 130(7) of the Patents Act further states that
[section] 1(1) . . . [is] so framed as to have, as nearly as practicable, the same [effect] in the United Kingdom as the corresponding provisions of the European Patent Convention, the Community Patent Convention and the Patent Co-operation Treaty have in the territories to which those Conventions apply.
Id. at 796.
179. Decision of the High Court of Justice, Patents Court (Merril Lynch's application) (1987), *reprinted in* 2M SINNOTT & COTREAU, *supra* note 166, at EPD-820.
180. *Id.*
181. *Id.* at EPD-822.
182. *Id.* at EPD-825.
183. T26/86 (KOCH AND STERZEL/X-ray apparatus) O.J. EPO 1988, 19, *cited in* PATERSON, *supra* note 28, at 317.
184. *Id.* at 318.

words, it does not prohibit the patenting of inventions consisting of a mix of technical and non-technical elements."[185]

In the case of IBM's patent application,[186] the patent application claimed a method of analyzing and displaying the events occurring in a text processing system in a computer. The EPO Board of Appeal held that the application was valid. There was no disclosure of any special hardware needed by the embodiment of the patent; it only required a combination of known hardware and new software. The Board of Appeal stated that a display of the internal conditions of an apparatus or system was a "technical problem."[187] The Board argued that even though the claimed idea resided in a computer program, the claim was directed toward the solution of a technical problem, and therefore the patent application was not seeking protection for the software as such within the meaning of EPC 52(2)(c) and (32).[188] This use of the "technical problem" test sidesteps the issue of patentability.[189] It would seem that the Board had already decided that a useful computer program could be patented. Having classified the patent to its liking, the Board found it easy to issue an opinion consistent with the classification. A European inventor can win the battle if his patent is acknowledged as a solution to a "technical problem." This backdoor method of patenting a computer program eviscerates EPC 52(3). Moreover, *New Scientist* magazine claims that the EPO is purposefully helping inventors get around the statutory bar by suggesting that they can get a patent if the claimed invention solves a technical, commercial problem.[190] The EPO has already granted approximately 11,000 software related patents,[191] and its approach to software patents has been copied by the national patent offices of Germany, the Netherlands, Great Britain, and Sweden.[192] In light of these facts, the only remaining effect of EPC 52(3) is that one cannot unartfully

---

185. *Id.*

186. T22/85 (IBM/Document abstracting and retrieving) O.J. EPO 1990, 12, *reprinted in* 20 JOHN P. SINNOTT & WILLIAM J. COTREAU, EPD-1900, EPD-1901 (1995).

187. *Id.* at EPD-1902.

188. *Id.*

189. "Instead of rejecting software patents out of hand, the examiners apply the 'problem and solution' test. If the inventor uses technical know-how to solve a commercial problem, the EPO grants a patent." Barry Fox, *Patents*, NEW SCIENTIST, July 29, 1995, at 21.

190. *Id. (quoting* the EPO's 1994 Annual Report, see *infra* note 191).

191. Patenting Computer Software 1994 Annual Report, European Patent Office [hereinafter EPO's 1994 Annual Report] (copy on file in the IND. INT'L & COMP. L. REV. office).

192. *Id.*

claim a computer program standing alone.[193] This is reminiscent of the conditions in the United States before *In re Alappat*.

IV. DISCUSSION OF CURRENT UNITED STATES LAW

A. *The Proposed Guidelines*

On June 2, 1995, the PTO published "Request for Comments on Proposed Examination Guidelines for Computer-Implemented Inventions."[194] The PTO Guidelines were issued in response to recent changes in case law.[195] The purpose of the PTO Guidelines is to stimulate feedback which will be used to measure the acceptability of the proposed changes. If adopted, the PTO Guidelines will bring Patent Office examination procedures into alignment with court rulings. The PTO Guidelines were also published to clarify the PTO's position on the patentability of software.[196] If adopted, the PTO Guidelines will be used by patent examiners for determining the patentability of computer-related patent applications. As of yet, no rulings have been made on a patent application which relies on the PTO Guidelines.[197]

The PTO Guidelines initially set forth three ways a software patent can be specified in a patent claim. The first claim method is as a "machine."[198] To classify an item as a machine, the claim must be directed towards "a computer or other programmable apparatus" controlled by a computer program or software.[199]

---

193. "[T]he new Guidelines of the European Patent Office state that 'patentability (of [the] subject-matter [of a patent application]) should not be denied merely on the ground that a computer program is involved in its implementation.'" Jack E. Brown, *Recent International Trends in the Legal Protection of Computer Software*, 2 J. L. & TECH. 167, 169-170 (1987) (quoting Gall, *European Patent Office Guidelines 1985 on the Protection of Inventions Relating to Computer Programs*, COMPUTER L. & PRAC., Sept.-Oct. 1985, at 6).

194. *See* PTO Guidelines *supra* note 34.

195. *Id.* at I(A).

196. Legal Analysis to Support Proposed Examination Guidelines for Computer-Implemented Inventions, at I [hereinafter Legal Analysis] (October 3, 1995). File "swanal.txt" *available in* the Internet at ftp.uspto.gov::/pub/software (copy on file in the IND. INT'L & COMP. L. REV. office).

197. In the rehearing of *Trovato*, the CAFC granted the petitioner's request. The previous judgment of the CAFC was vacated, along with the decision of the Board of Patent Appeals. The case was remanded to the PTO for reconsideration. The CAFC made its decision based on the issuance of the PTO Guidelines, but Trovato's patent application did not rely on the PTO Guidelines, as it in all likelihood was drafted long before the PTO Guidelines were issued. *See In re* Trovato, 60 F.3d 807 (1995).

198. PTO Guidelines, *supra* note 34, at I(B)(1)(c)(i).

199. *Id.* at I(B)(1)(c)(i).

The second claim classification is an "article of manufacture."[200] The PTO Guidelines define an article of manufacture as having two elements. The first element is a computer-readable storage medium.[201] This includes, but evidently is not limited to a floppy disc, CD-ROM, or a "memory device."[202] The memory device is required to "impart the functionality represented by the data onto a computer."[203] The key word here is functionality. The memory device must contain more than just data, it must make the computer function. The second element is the configuration of the storage medium. The configuration must cause the using computer to operate in a specific and predetermined manner.[204] The two elements combine to form a storage medium featuring a specific structure and function. For these machines and articles of manufacture, the patent claims are required to define discrete physical structures.[205] These structures "may be comprised of hardware or a combination of hardware and software."[206]

The third type of claim classification is a claim of the algorithm as a "process."[207] For a software patent to qualify as a process, the claim must recite a series of specific operational steps to be performed on or with the aid of a computer.[208] It is implied in this specification that a physical machine is part of the claimed process.

The PTO Guidelines also specify non-statutory subject matter for computer software. The principle behind these non-statutory categories is that an invention must represent a practical application of an idea to be statutory.[209] The PTO states that it is not fear of preemption but rather the

200. *Id.* at I(B)(1)(c)(ii).

201. *Id.* at I(C)[1](1).

202. The guidelines do not specify what is meant by a memory device. The Legal Analysis states that an article of manufacture "will typically be a component of a specific computer, such as a logic circuit or a computer memory." Legal Analysis, *supra* note 196, at III(B)(2)(a)(i). A memory device could mean any Random Access Memory (RAM) or Read-Only Memory (ROM), since RAM and ROM are composed of a multitude of memory logic circuits.

203. PTO Guidelines, *supra* note 34, at I(C)[1](2). The PTO also tells inventors how to define a computer memory. A claim must identify "the physical characteristics of the memory (e.g., a logic circuit or a storage medium), and the functionality of the memory." Legal Analysis, *supra* note 196, at (III)(B)(2)(a)(i). A computer memory can be defined in a claim in three ways. The first way is as a logic circuit formed by the loading of software onto a computer. *Id.* The second way is as a computer memory that is defined by its purpose or organization. *Id.* The third way of defining a computer memory is by claiming the physical arrangement of the memory that results when the computer program is executed. *Id.*

204. PTO Guidelines, *supra* note 34, at I(C)[1](2).

205. Legal Analysis, *supra* note 196, at (II)(C).

206. *Id.*

207. PTO Guidelines, *supra* note 34, at I(B)(1)(c)(iii).

208. *Id.*

209. Legal Analysis, *supra* note 196, at (III).

"practical application" requirement that prevents abstract ideas, laws of nature, and natural phenomena from being patentable.[210] There are four specific proscriptions. The first is an arrangement of *data* which is not tied to any physical element.[211] The second type of non-statutory subject matter is a storage medium which contains *data* that is a creative or artistic expression.[212] This prohibits any pictures or music on a disc or CD-ROM from being patentable. These two prohibitions of data do not apply to computer programs, which are not merely data but are also data structures. The third type of non-statutory subject matter is a data structure[213] which is not tied to any physical element. This prohibition is a reinforcement of the mandate that a software "article of manufacture" must include 1) some physical memory substrate in which the software resides; 2) a physical apparatus if the software is claimed as a "machine"; or 3) a series of steps to be performed on a computer if the software is claimed as a "process." The fourth type of non-statutory subject matter is a process that only manipulates abstract ideas or concepts.[214] This proscription illustrates the often quoted proviso that a mathematical formula cannot be patented.[215] This prohibition is necessary because the PTO's definition of a process is wide open. The PTO evidently felt that it needed to narrow its definition of a process as a "series of operational steps."

B. *Comparison of the Guidelines to Legal Precedent*

The PTO Guidelines were designed to follow all applicable court precedent. But the Guidelines significantly broaden precedent in defining a "process." The PTO Guidelines do not present a new definition, but the Legal Analysis extends the definition of a process to any "electrical signal representing data corresponding to a physical object or physical activity."[216] The first significant computer software precedent was established in *Benson*. The PTO Guidelines specifically point out that a claim that does nothing more than convert one set of numbers into another set of numbers is non-

---

210. *Id.*
211. PTO Guidelines, *supra* note 34, at I(B)(1) (emphasis added).
212. *Id.*
213. The PTO's legal analysis defines a data structure as "[t]he relationship that exists among the ordered data elements." Legal Analysis, *supra* note 196, at III(B)(1)(a). At its most basic level, a computer program is just a sequence of numbers. The order and structure of the numbers is what give the program its meaning, just as the order of the alphabetic characters in a book make a story.
214. PTO Guidelines, *supra* note 34, at I(B)(1).
215. *See, e.g.*, Gottschalk v. Benson, 409 U.S. 63, 67 (1972).
216. Legal Analysis, *supra* note 196, at (III)(B)(2)(b)(i).

statutory.[217] This preserves the holding in *Benson*,[218] but as a reinforcement of the prohibition against patenting laws of nature, it is unnecessary in light of the fourth prohibition which disallows processes that only manipulate abstract ideas or concepts. Nonetheless, the preemption test of *Benson* has been eliminated in favor of a test for practical application. The decision of the CAFC in *In re Warmerdam* stated that patent claims as a whole must go beyond simply manipulating abstract ideas.[219] The PTO Guidelines embrace this holding. The CAFC in *In re Trovato* held that a patent application should recite some hardware to be statutory.[220] The PTO Guidelines maintain this holding but expand the definition of hardware. The PTO Guidelines view inclusion of a computer memory device to be a sufficient hardware disclosure, but it seems unlikely that the *Trovato* court would find it sufficient.

V. CONCLUSION

The practical effect of the PTO Guidelines is that now an inventor can patent a computer program as part of a process, part of a machine, or on its own, if it is claimed as a part of a computer memory. In the near future a computer program on a floppy disc may be patentable standing alone. A computer program that is claimed without any physical structure cannot yet be patented, even though the software would be patentable if claimed in conjunction with an appropriate physical structure. Ultimately the PTO Guidelines are nothing more than a codification of the holding in *Alappat*. But puzzling questions remain. For example, it is unclear why a computer program must be claimed as a machine when a "machine" is understood to be necessary for its use. Perhaps this is a concession to those who maintain that patents should not be granted on software.

Changes in patent law are ultimately like changes in other areas of law; they occur slowly and in small increments. These changes are evolutionary and are not a change of direction. The attitude towards software patents in the United States has been slowly changing from one of suspicion to grudging acceptance. If the new PTO Guidelines are implemented, computer programs will be almost as patentable as other forms of technology. In contrast to the attitude of the United States, the EU seems to have a reactionary attitude toward software patents. There has been very little indication that their attitude might change. Copyright is still a widely accepted form of software protection in the EU. The lack of official

---

217. PTO Guidelines, *supra* note 34, at I(C)[5].
218. *Benson*, 409 U.S. at 67 (holding that a patent application for converting binary-coded decimal numbers into pure binary numbers was not patentable subject matter).
219. *In re* Warmerdam, 33 F.3d 1354, 1360 (Fed. Cir. 1994).
220. *Trovato*, 42 F.3d at 1382.

encouragement in the EU toward software patents may become very telling in the future. It seems highly likely that the current differences between the United States and the EU will work in the United States' favor. The availability of software patents will give American software developers added protection and this will encourage growth in the domestic software industry. Although the processes for obtaining a patent for a computer program in the United States and Europe are still somewhat similar, the difference between the processes is growing. The largest difference currently is seen in the official attitudes of the respective patent offices. The small change made by the United States could lead to the acceptance of software as patentable subject matter all over the globe.

Gregg Jansen[*]