Santa Clara Law Review

Volume 23 | Number 4 Article 2

1-1-1983

Patent Protection of Computer Software - Practical Insights

James A. Sheridan

PATENT PROTECTION OF COMPUTER SOFTWARE—PRACTICAL INSIGHTS

James A. Sheridan

Whether or not computer software is patentable has been one of the most controversial issues in intellectual property protection over the last twenty years. It now appears that the United States Patent Office will grant patents on some software.[1] This article examines the issue of patentability of computer software and some of the advantages and disadvantages of patent protection.

It is important to remember that there are several facets to the question of whether computer software is patentable. This discussion is concerned only with the issue of whether computer software is "patentable subject matter" under the patent laws.[2] However, in addition to being proper subject matter, the subject of a patent claim must also be shown to be new[3] and unobvious.[4] The patent laws further impose the requirement that the invention be adequately disclosed so that anyone else working in the same field of technology could reproduce the claimed invention.[5]

The distinction between what is patentable and what is unpatentable under 35 U.S.C. section 101 was drawn by the United States Supreme Court over 100 years ago. In *O'Reilly v. Morse*[6] the Court stated:

> The mere discovery of a new element, or law, or principle of nature without any valuable application of it to the arts, is not the subject of a patent. But he who takes this new element or power, as yet useless, from the laboratory

---

1. MANUAL OF PATENT EXAMINING PROCEDURE, § 2110 (1983).
2. 35 U.S.C. § 101 (1976). § 101 provides: "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."
3. 35 U.S.C. § 102 (1976).
4. 35 U.S.C. § 103 (1976).
5. 35 U.S.C. § 112 (1976).
6. 56 U.S. (15 How.) 62, 131 (1853) (Grier, J., dissenting).

> of the philosopher, and makes it the servant of man; who applies it to the perfecting of a new and useful art, or to the improvement to one already known is the benefactor to whom the patent law tenders its protection.[7]

In the context of a discussion of computer software, the issue raised in *O'Reilly* is whether a computer program which performs an algorithm[8] is the proper subject of a patent. The Supreme Court has finally recognized that when an algorithm does more than represent a scientific principle or law of nature and instead becomes a vehicle for communicating a solution to a complex problem in a particular environment, then its use can be the basis for patent protection.[9]

Relatively few cases have been decided by the federal district courts on the patentability of computer software.[10] In examining those cases that have been decided, it is significant that in the Patent Office the final court of review has always been almost exclusively the Court of Customs and Patent Appeals (CCPA).[11] This court has consistently maintained that no basis exists for treating a computer implemented process differently from a process performed by any other machine system when the issue of statutory subject matter is considered. The CCPA has focused its inquiry only on whether a claim reciting or covering a software implemented process is attempting to wholly preempt the use of the algorithm or is intended to cover only a method of calculation in a given environment.[12]

The significance of the CCPA's consistent viewpoint is that in 1982 it became part of the Court of Appeals of the

---

7. *Id.* at 132.

8. An algorithm is a step-by-step procedure for solving a problem. WEBSTERS NEW COLLEGIATE DICTIONARY 28 (1979).

9. Diamond v. Diehr, 450 U.S. 175 (1981). The Court stated: "When a claim containing a mathematical formula implements or applies that formula in a structure or process which, when considered as a whole, is performing a function which the patent laws were designed to protect . . . then the claim satisfies the requirements [of patent law]." *Id.* at 192.

10. Pursuant to 28 U.S.C. § 1338, the federal district courts have exclusive jurisdiction over patent cases.

11. Appeal to the CCPA is pursuant to 35 U.S.C. § 141 (1976). The other choice, rarely used, is a trial de novo in the U.S. District Court for the District of Columbia. This is not a true appeal, but a civil action to obtain a patent. 35 U.S.C. § 145 (1976).

12. Application of Prater, 415 F.2d 1378 (C.C.P.A. 1968), *modified on reh'g*, 415 F.2d 1393 (C.C.P.A. 1969).

Federal Circuit. It now has exclusive jurisdiction in all appeals from the federal district courts of all cases related to the validity and infringement of patents.[13] Therefore, in a proceeding to enforce a patent granted on computer software, an appeal must be taken to the CCPA. Based on the history of CCPA decisions over the last ten years, this court will almost certainly have a strongly favorable view toward upholding patents based on computer software, provided that the software is properly disclosed and claimed.

The CCPA's approach constitutes a considerable change from the Supreme Court's attitude of the early 1970s. At that time it was generally believed that software implemented or algorithm based inventions were unpatentable. The Supreme Court first examined the issue of patenting complete software in *Gottschalk v. Benson*[14] which involved a method for converting from binary coded decimal numbers into pure binary numbers within a computer.[15] The claim, as written, would have wholly preempted the use of this algorithm within any computer, and the Court held that the claim to this formula was akin to the recitation of a law of nature. The court stated correctly that: "The formula involved here has no practical application except in connection with the digital computer, which means that if the judgment below is affirmed, the patent would *wholly preempt* the mathematical formula and a practical effect would be a patent on the algorithm."[16]

The Court further stated that the patentability of computer software was a matter for congressional legislation and that the Court should not attempt to intervene in the matter.[17]

In fact, a Presidential Commission formed in 1965 had recommended against patent protection for computer programs.[18] Legislation to that effect was proposed in 1967; however, because of considerable opposition, it was withdrawn and never reintroduced.[19]

---

13. 28 U.S.C. § 1292 (1982).

14. 409 U.S. 63 (1972).

15. The purpose of this program was to convert telephone numbers into binary form which was necessary for telephone interconnection. *Id.*

16. 409 U.S. at 71.

17. *Id.* at 73.

18. PRESIDENT'S COMMISSION ON THE PATENT SYSTEM (1966).

19. H.R. 5924, 90th Cong., 1st Sess. (1967); S. 1042, 90th Cong., 1st Sess. (1967).

The Supreme Court next addressed the issue of the patentability of software in *Parker v. Flook*.[20] The claim in *Parker* described a method of updating alarm limits. A mathematical algorithm was used to compute the value of environmental limits in a catalytic hydrocarbon conversion process. In essence, the method consisted of three steps: (1) a step of measuring the present value of selective process variables (temperature); (2) an intermediate step which used the algorithm to calculate an updated alarm limit value; and (3) a final step in which the actual alarm limit is adjusted to the updated value.[21]

The Supreme Court found the software program unpatentable. The Court stated that this case must be considered as if the principle, the formula for calculating the updated alarm limits, was well known and that the plaintiff had first invented a mode of applying it. However, since the plaintiff did not even describe any specific mode of applying the calculation, there was no patentable subject matter.[22] Therefore, this application of *Flook* was essentially attempting to preempt a method of calculating without providing any supporting connection to a real-world environment.

Finally, in 1981 the Supreme Court in *Diamond v. Diehr*[23] allowed a patent in which the novel features clearly resided in the calculating algorithm, but the algorithm was disclosed and claimed in a real-world operating environment. The invention in *Diehr* involved the process of constantly measuring the actual temperature inside a rubber curing mold using a known equation. The temperatures were then fed into a computer which repeatedly recalculated the cure time by use of the same equation. At the appropriate time, the computer would signal a device to open the press.

---

20. 437 U.S. 584 (1978).

21. *Id.* at 585.

22. The Court stated:

> The patent application does not purport to explain how to select the appropriate margin of safety, the weighting factor or any of the other variables of the equation. Nor does it purport to contain any disclosure relating to the chemical processes at work, the monitoring of process variables, or the means of setting off an alarm or adjusting an alarm system. All that it provides is a formula for computing an updated alarm limit.

437 U.S. at 586.

23. 450 U.S. 175 (1981).

The applicants contended that the processes involved in the continuous measuring of the temperature inside the mold cavity, the feeding of this information to a digital computer which constantly recalculated the cure time, and the signaling by the computer to open the press were all new in the art.[24] The Court held that the equation itself was not patentable, in isolation, but when the process for curing rubber was devised which incorporates this equation in a more efficient manner, that process is not barred from patentability by section 101.[25]

It is worth noting that the decision in the *Diehr* case was the third pro-patent decision of the Supreme Court in less than one year.[26] Prior to 1981 the Supreme Court had not upheld a single patent since *United States v. Adams*[27] which was decided in 1966. Throughout the intervening time the Court, in striking down patents, had repeatedly stressed the "nation's deep seated antipathy to monopolies."[28] In the cases decided in 1981, the opinions discussed the positive factors of encouraging invention and research and development which result from the patent system.[29]

It is further worth noting that thanks to Apple Computer, Inc., Atari, Inc., and similarly situated companies, the computer had moved from the laboratory and university and into the homes of people throughout the country, making computer innovation and invention a much less awesome subject with which to contend.

Diehr limited the scope of the previous decisions in *Gottschalk* and *Flook*.[30] Now, only those claims which recite a mathematical formula in the abstract appear to be excluded

---

24. *Id.* at 179.

25. *Id.* at 184.

26. *See* Diamond v. Bradley, 450 U.S. 381 (1981); Diamond v. Chakrabarty, 447 U.S. 303 (1980).

27. 383 U.S. 39 (1966).

28. *See, e.g.*, Deep South Packing Co. v. Laitram Corp., 406 U.S. 518, 530-31 (1972); Graham v. John Deere Co., 383 U.S. 1, 7-10 (1966).

29. *See, e.g.*, Diamond v. Chakrabarty, 447 U.S. at 307.

30. 450 U.S. at 185-87. However, the similarities between what was disclosed and claimed in *Diehr* and what was claimed in the *Flook* case decided four years earlier are remarkable. It would appear that the only reasonable rationale for the different results in the two cases is the lack of a close relationship in *Flook* between the claimed calculation and the process to be controlled. That is, *Flook* did not claim steps of terminating or adjusting the process of catalytic conversion, nor did the claim specifically describe the process which was to be controlled or the interaction between process and monitoring computer.

from patent protection.[31]

While the Supreme Court was agonizing over the basic issue of the patentability of software, the CCPA, whose decisions consistently favored patentability, was developing a two-step test for analyzing whether a claim involving computer programming or an algorithm constitutes potentially patentable subject matter under 35 U.S.C. section 101.[32] As a first step, each claim must be analyzed to determine whether an algorithm is either directly or indirectly recited. Thus, even if the claim at issue does not directly recite a mathematical algorithm or calculation, reference will be made to the patent application specification to determine whether the claim language is intended to cover a mathematical calculation, formula or equation. Only if the answer to this first question is yes must the second step of the test be taken.

The second step is to determine whether the claim would preempt the algorithm's use by anyone for any purpose. If so, the claim will be held unstatutory under 35 U.S.C. section 101. However, if the claim recites a calculation which is imminently related to the environment in which the invention is used and controls a process or transforms an article, it should be protectable by patent.

A threshold question which might be considered is whether there is significant use of the results of the mathematical calculation, such as in *Diehr* where the result of the calculation was used to actually open the press in which the rubber was being formed, or whether the calculations merely involve one set of numbers being computed from a different set of numbers by means of a mathematical computation.[33]

Some of the recent cases decided by the CCPA provide examples of the application of this two-part test. In *In re Toma*[34] the CCPA examined a claimed method of utilizing a digital computer for translation from a source natural language to a target natural language (from Russian to English, for example). The method involved three phases: (1) a dictionary lookup phase to establish the target language meaning

---

31. 450 U.S. at 191. Moreover, the Court notes that "insignificant postsolution activity will not transform an unpatentable principle into a patentable process." *Id.* at 191-92.

32. In re Freeman, 573 F.2d 1237 (C.C.P.A. 1978).

33. In re Gelnovatch, 595 F.2d 32 (1979).

34. 575 F.2d 872 (C.C.P.A. 1978).

of each word in the source test; (2) a syntactical analysis for identifying information from the inflection of the word and position of the word in the source; and (3) a synthesis phase for taking the information thus generated and forming a sentence in the target language. Although the method clearly used information stored in a digital computer, an analysis of these three steps does not disclose any use of equations to perform the claimed steps of the method and, thus, the first step of the two-part test has not been met. Therefore, one need proceed no further in concluding that this covers statutory subject matter under 35 U.S.C. section 101.

In re Pardo[35] provides another example of a patent claim where the first question of the two-part test was answered negatively and ultimately a patent was granted. The invention involved a method for automatically rearranging random formulae for sequential execution by a computer. Such rearrangement was necessary where a user provided formulae to a computer in an order which could not be executed because one step presented early in a sequence could not be performed until the results of later steps were attained. The invention was designed to rearrange the order of the formulae as presented to the computer by the user so that the computer could execute the operations. The CCPA held that the invention did not recite a mathematical formula, calculation or algorithm. The fact that a computer controlled according to the invention was capable of handling mathematics was irrelevant to the question of whether a mathematical algorithm was recited by the claims. Accordingly, the court held that the claims met the test for statutory subject matter.

In re Sarker[36] and *In re Gelnovatch*[37] provide models of inventions which were not deemed patentable under the two-part test. Both of these cases related solely to the use of mathematical models for design simulation functions. In *Sarker* the invention was a model for designing ship channels or the like. In *Gelnovatch* the model was for an optimal design process for the design of microwave circuits. In both cases purely mathematical functions were being carried out on input numbers which were modified to optimize the output of

35. 684 F.2d 912 (C.C.P.A. 1982).
36. 588 F.2d 1330 (C.C.P.A. 1978).
37. 595 F.2d 32 (C.C.P.A. 1979).

the mathematical models. As such, the process as carried out did not operate on any real numbers tied to a defined real environment. The claims in both cases were found to be mathematical formulae under step one and, because under step two they wholly preempted the use of the underlying algorithms, they were not patentable.

In *In re Johnson*[38] three related patent applications directed to methods for removing undesired noise from seismic data were considered.[39] The CCPA held that each of the three inventions met the first step of the test in that the claims covered a calculation, formula or equation for reducing noise or enhancing digital data. However, the claims also were found to cover a process which produced a new product comprising new noiseless seismic traces recorded on a record medium, rather than mere mathematical values. This decision has two significant aspects: (1) the process did not operate on a physical entity as such, but on a recording of electronic signals; and (2) the output was not the control of a process, but rather a recording on a physical medium. Nevertheless, because the process was operated on a set of real values to produce enhanced signal recordings in a new and not obvious way (even though old mathematical equations were used) the process was held to constitute patentable subject matter.

In re Abele[40] is of interest because it furnishes an example of how the manner in which claims are drafted may affect whether a patent is granted. The claims were directed to a method of analyzing the data in a computerized axial tomography (CAT) scanner. The method calculated the data and displayed results. A claim which recited nothing more than the steps involved in the calculation and display of data was found to comprise nonstatutory subject matter under the two-

---

38. 589 F.2d 1070 (C.C.P.A. 1978).

39.

> In seismic processing an acoustic or seismic wave energy source is positioned on the surface of the earth and an acoustic energy impulse is generated. The secondary waves which reflect from different layers in the surface in the earth are recorded and stored in digital form. These waves are then analyzed to disclose the location of subterranean earth structures. The invention in each of the three applications in these appeals dealt with the removal of unwanted seismic components or noise present in the recorded seismic data.

589 F.2d at 1071.

40. 684 F.2d 902 (C.C.P.A. 1982).

part test—the claims covered a method of mathematical calculation and wholly preempted the use of the calculating algorithm.[41] However, a second claim differed in that it recited that the calculations were performed on data which had passed through the body of a person under examination by the scanner. This claim was deemed to recite patentable subject matter and did not preempt the use of the underlying algorithm from *all* use, but rather only preempted its use in a specific environment, that of CAT scanners.[42]

As the foregoing discussion indicates, guidelines now exist to determine whether a claim consists of patentable subject matter under 35 U.S.C. section 101. In deciding whether to file for patent protection, one must weigh the broad scope of protection afforded by patent against the time needed to issue a patent and the need for full disclosure to the public.

Time could be a key factor in deciding whether or not to apply for a patent. Patents take three to five years to issue. If the software is expected to be of a relatively short life, patent protection is probably not worthwhile. However, if the underlying algorithm of the software is expected to remain valuable for a number of years, patent protection may be worthwhile. This is especially true if the application or operating systems will undergo modification with time, while the basic functions of the software implemented system remain the same.

The second key factor is the necessity of making an adequate disclosure to support the application. Under 35 U.S.C. section 112, the disclosure must be adequate to inform one skilled in the art how to make and use the invention without undue experimentation.[43] It is generally conceded that to support a sufficient disclosure, at least a block diagram of the complete system and a flow chart of the software to be uti-

---

41. *Id.* at 908.

42. *Id.*

43. 35 U.S.C. § 112 provides:

> The specification shall contain a written description of the invention, and of the manner and process of making and using it, in such full, clear, concise, and exact terms as to enable any person skilled in the art to which it pertains, or with which it is most nearly connected, to make and use the same, and shall set forth the best mode contemplated by the inventor of carrying out his invention.
>
> 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.

35 U.S.C. § 112 (1976).

lized must be provided.[44] Thus, an applicant must consider whether, if only a flow chart of the system is provided, an inordinate amount of time would be required to generate the computer software. If the software could only be duplicated by a person of unusual skill in computer programming, and would require an inordinate period of time for debugging in order to make it work properly, then additional information, such as the software source or object code must be disclosed.[45]

Although this disclosure requirement seems to weigh heavily against applying for a patent, it should be remembered that a patent remains secret until it is issued. Therefore, patent protection is not incompatible with trade secret protection at least for the first three to four years of the use of the software. Only when it is time for the patent to issue must the decision be made on whether patent protection or trade secret protection affords the best hope of maintaining competitive advantage. Further, although the patent laws require that on the date a patent application is filed the best known embodiment of the software be disclosed, that embodiment need not be updated while the patent application is pending. Therefore, as long as a working embodiment is supplied to the Patent Office at the time the case is filed, the improvements which may be incorporated in software between the time the application is filed and the time it issues need not be disclosed in the issued patent.

Given the two considerations of time and necessity of adequate disclosure, it might appear that the use of patent as a means of protection has some serious drawbacks. The primary advantage of patent protection lies in its broad scope. The scope of protection afforded by patent is probably broader than that afforded by either copyright or trade secret. Patent protects an exclusive right to use or license, whereas copyright only protects an exclusive right to copy, and trade secret only protects an exclusive right to use. Moreover, the cost of maintaining protection is almost nil, a factor which is not true of

---

44. *See* In re Ghiron, 442 F.2d 985 (C.C.P.A. 1971).

45. Whether the disclosure requires a disclosure of source code or object code is still an open question. The disclosure of object code, together with a definition of the specific type of processor for which the software is written, would certainly allow a person of skill in the computer arts to duplicate the invention. This conclusion would allow the person who seeks a patent to avoid disclosure of the source code.

copyright or trade secret.[46] And, in practice, reverse engineering of a product on the market is not a successful means of avoiding a patent, as it may be with either copyright or trade secret.

In conclusion, the protection of computer software continues to pose many challenging questions. The Supreme Court's decision in *Diamond v. Diehr* appears to have resolved the question of patentability of software. And the *Freeman* test adopted by the CCPA offers a consistent standard by which patent claims can be measured. Consequently, patent has become an increasingly viable means of protecting this form of intellectual property. This presents a challenge for the practitioner. He must be prepared to consider all of the traditional means available for the protection of ideas—copyright, trade secret, patent and contractual arrangements—and determine which best suits his client's needs.

While patent protection has its limitations, it appears to be very well suited for the protection of computer software due to the exclusive rights it encompasses. Patent has the added advantage of allowing inventors to reap the rewards of their ideas while benefiting society by the dissemination of those same ideas.

46. A discussion of the differences between patent, trademark and copyright protection is beyond the scope of this article. However, see Appendix A for a general overview of the differing characteristics of these forms of protection for intellectual property. Appendix A was adapted from Reiling and Lester, *Marketing Software Products*, 8 ALPA Q.J. 294, 298, 300 (1980).

Appendix A

COMPARISON OF LEGAL FORMS OF PROTECTION

CONSIDERATION	COPYRIGHT	TRADE SECRET	PATENT
National Uniformity	Yes	No	Yes
Protected Interest	Fixed expressions of author	Ideas and expressions	Invention
Scope of Protection	Exclusive right to reproduce, prepare derivative works, publicly display and publicly perform.	Exclusive right to use	Broadest, excludes others from making, using, selling
Effective Date of Protection	Fixation of work in sufficiently permanent and tangible form	Use in Business provided that subject matter is guarded from public disclosure.	Issue of patent successful prosecution of patent application
Cost of Obtaining Protection	Small	Moderate	Moderate
Term of Protection	Life of author plus 50 years or 75 years	Possibility of both perceptual protection and termination at any time	17 years
Cost of maintaining protection	Small	Significant	Nil
Cost of Enforcing Rights against violators		Moderate	High High
Protection Lost by-	Gross neglect	Public Disclosure	Unsuccessful validity or misuse litigation
Internationally	Often	Not generally	Often, but foreign filing may disclose before U.S. rights perfected
Execution of Software Products Protectable	No	Yes	Yes
Suited to wide-scale distribution	Yes	No	Yes