INTELLECTUAL PROPERTY PROTECTION FOR DIGITAL SYSTEMS

David P. Krivoshik, Esq., PE
PatentLaw@acm.org

presented at
DIMACS Workshop on Management of Digital Intellectual Property
April 17 - 18, 2000
The Center for Discrete Mathematics and Theoretical Computer Science
DIMACS Center, Rutgers University, Piscataway, NJ

Abstract

Traditionally, computer software, a process based on a mathematical formula, and a method of doing business were three types of intellectual property that were excluded from patentability. The courts have referred to these exceptions as the "mathematical algorithm" exception and the "business method" exception. The U.S. Supreme Court has upheld Federal Court decisions essentially indicating that computer software, a process based on a mathematical formula, and a business method are patentable subject matter. This has opened the door to new and powerful ways to protect these kinds of intellectual property. This trend leads to the interesting situation in which software can be protected simultaneously by patent, copyright, and trade secret laws. A patent can protect a software idea, while copyright and trade secret laws can protect the details of the software as an unpublished work. This multifaceted legal protection can provide significant financial and competitive advantages for the software developer, as well as establish barriers.

Intellectual Property Protection for Digital Systems

By David P. Krivoshik, Esq., PE

Introduction

Computer software is no longer restricted to the domains of large business operations and educational environments. Today, computer software is embedded in the household appliances we use and even in the automobiles we drive. On the information superhighway, computer software provides the underlying foundation to accessing all content over the Internet through the World Wide Web. Even the presentation of digital content is controlled by software.[1] Traditionally, computer software, a process based on a mathematical formula, or a method of doing business were excluded from patent protection and protected only to a limited extent by copyright[2] and trade secret[3] laws. The U.S. courts have referred to these exceptions as the "mathematical algorithm" exception and the "business method" exception.

Background

The application of traditional intellectual property protection to computer software raises a debate over the application of established legal protection to new technologies. However, computers and computer software are not recent developments. In various forms, the Egyptians, Greeks, Romans, and Chinese used the abacus (a digital computer). In 1642, a Frenchman in his early twenties, Blaise Pascal, invented a machine to aid his father in his work. Pascal's "arithmetic machine" worked very much like today's odometers. The numbers 0 through 9 were printed on the edges of a row of wheels. When a wheel made a complete turn from 0 through 9, a small notch caused the next wheel to the left to move up one number. The machine could only do addition or subtraction operations. In 1801, another Frenchman, Joseph Jacquard, invented a new type of loom for weaving cloth that revolutionized the weaving industry in the early part of the century. Punched cards were used to control the operation of the loom. Yarn was pulled up through the holes in the cards

---

[1] Java Applets or simple HTML code is used to present multimedia content for access by a Web browser.

[2] Copyrights protect the expression of an idea. Copyright is a form of protection provided by the laws of the United States (title 17, U.S.C.) to the authors of "original works of authorship," including literary, dramatic, musical, artistic, and certain other intellectual works. This protection is available to both published and unpublished works. Protection exists in the United States without the necessity for registration in most cases. U.S. copyright protection is secured automatically when the work is created. A work is created when it is fixed in a copy for a first time.

[3] A trade secret protects an idea or an expression of an idea by keeping the idea or expression of the idea secret. Among the factors often used by U.S. courts to determine whether or not an enforceable trade secret exists are:

(1) whether or not the information is known outside the business of the trade secret holder (not known is good);
(2) the extent to which information is known by employees and others in the business (less is better);
(3) the measures taken by the employer to guard the secrecy of the information (more is better);
(4) the value of the information to the employer and to his/her competitors (higher is better);
(5) the ease or difficulty with which the information could be properly acquired or duplicated by others (difficult is good); and
(6) the unfair advantage that would or has occurred because of the misappropriation of the trade secret (the more unfair the better).

to make the pattern. Jacquard's punched cards stored information. In 1834, an Englishman, Charles Babbage, devised most of the principles on which modern computers are based. His Analytical Engine was the first calculating machine that could truly be called a computer. The Analytical Engine had all four parts of a computer system: input, output, memory, and a central processing unit. All of the machines Babbage invented were, of course, mechanical, being combinations of gears, wheels, and levers, operated by steam. Unfortunately, a working model could not be built because the tools of that time simply could not accommodate his sophisticated design and Babbage died never knowing that his ideas would be widely acclaimed more than one hundred years later. During the nineteenth century, the United States government was faced with the cumbersome task of taking the census and tabulating the results. The number of years necessary to compile usable information made the information virtually obsolete by the time it was available. A young army engineer and inventor, Herman Hollerith, was assigned the task of making the census system more workable, and before the 1890 census he built the Tabulating Machine. Hollerith's invention borrowed the idea of punched cards from Jacquard. His machine used cards punched with a special code. Name, address, sex, home state, and more data were punched so the electric circuits would connect where there was a hole. The information was then automatically tabulated and stored. Hollerith's machine completed the census tabulations in five years' less time than the methods used in tabulating the previous census. After improving the machine for commercial use in accounting as well as statistical tabulating, Hollerith left the census office and founded the Tabulating Machine Company. His company, along with other companies, formed a conglomerate that eventually evolved into IBM. Descendants of the Hollerith machines continued to form the basis of IBM's business until the advent of the on-line computer era.

First U.S. Software Patent

With the judicially created mathematical algorithm exception, it would seem that software patents are only a recent development. But one of the possibly earliest software patents is U.S. Patent No. RE117 issued on June 13, 1848, to Samuel F. B. Morse. In the patent, claim number 5 reads:

> The system of signs consisting of dots and spaces, and of dots, spaces, and horizontal lines, for numerals, letters, words, or sentences, substantially as herein set forth and illustrated, for telegraphic purposes.

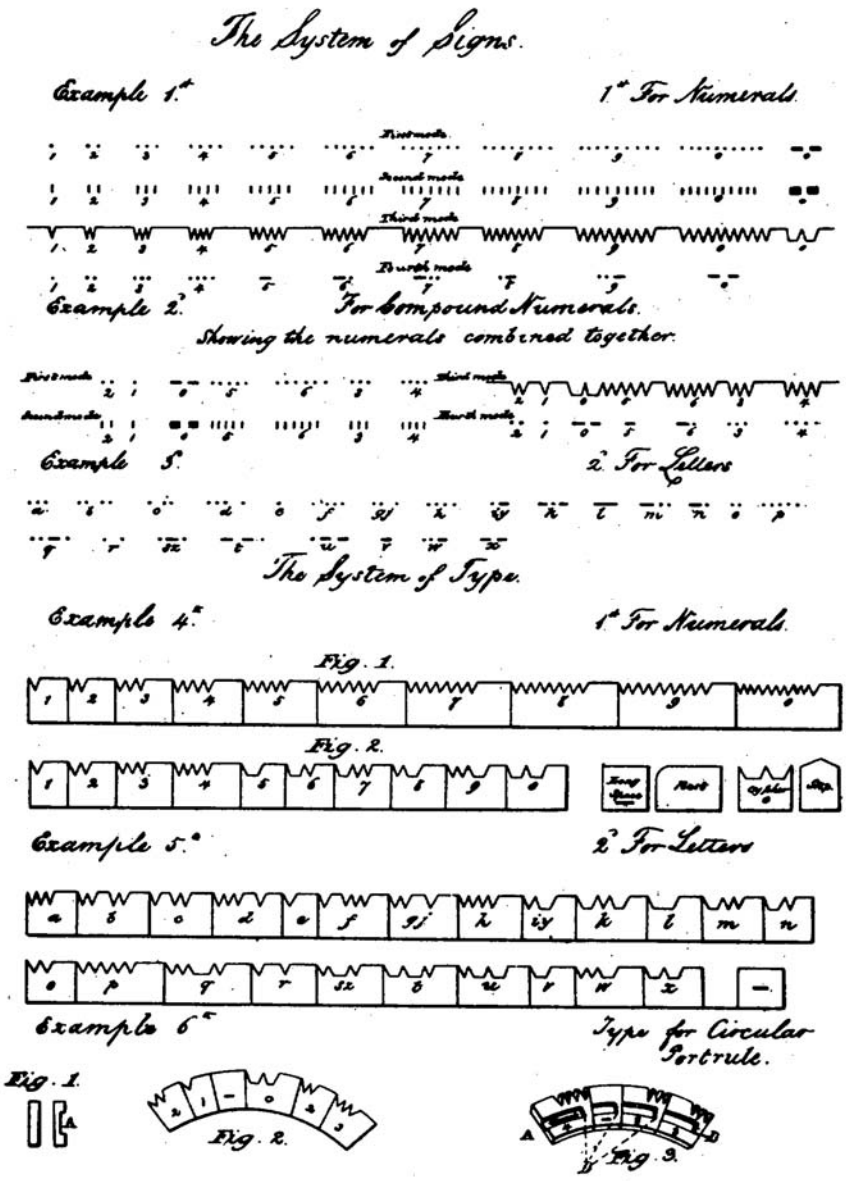
This is essentially what is now called and recognized by the U.S. Patent and Trademark Office as a "signal claim." Morse described and claimed a base three (dots, dashes and spaces) signaling system. The Supreme Court upheld the validity of the Morse patent. In spite of the early software patent issued to Morse, software was not typically patented or even considered patentable until very recently.

No. 117. REISSUED JUNE 13, 1848.

S. F. B. MORSE.

ELECTROMAGNETIC TELEGRAPH.

4 SHEETS—SHEET 1.



5. The system of signs consisting of dots and spaces, and of dots, spaces, and horizontal lines, for numerals, letters, words, or sentences, substantially as herein set forth and illustrated, for telegraphic purposes.

6. The system of signs consisting of dots and spaces, and of dots, spaces, and horizontal lines, substantially as herein set forth and illustrated, in combination with machinery for recording them, as signals for telegraphic purposes.

Software Patentability

In determining if a patent is obvious, prior art must be considered. With software, most of the prior art is non-patent art, which is quite difficult to search, rarely considered by the U.S. Patent Examiners, and often not considered until litigation. This results in weak and, in some cases, quite ridiculous patents. In considering the patentability of software, non-patent art must be carefully considered and addressed. A similar issue of non-patent art must be dealt with for methods of doing business[4, 5].

However, § 101 of the patent law reads:

> 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.

Any invention falling within one of the four stated categories of statutory subject matter may be patented, provided it meets the other requirements for patentability in §§ 102, 103, and 112.

Over time, the courts have expanded the interpretation of § 101, recognizing the growing importance and integral nature of software.[6]

In *In re Alappat*[7], the Federal Circuit held in 1994 that data mathematically transformed by a machine to produce a smooth waveform on a display constituted a patentable application of an abstract idea because it produced a "useful, concrete and tangible result"—the smooth waveform. And in 1992, in *Arythmia Research Technology Inc. v. Corazonix Corp.*[8], the court held that the mathematical transformation of electrocardiograph signals from a patient's heartbeat in order to determine the condition of the patient's heart constituted patentable subject matter.

In *State Street Bank & Trust Co. v. Signature Financial Group Inc*., for the first time, a mathematical algorithm that produces numbers as its only output was classified as constituting patentable subject matter if it produces a result that is "useful, concrete and tangible."[9]

---

[4] On March 29, 2000, USPTO director, Q. Todd Dickenson, announced a "Business Methods Patent Initiative: An Action Plan" to fine-tune the examination process for business-method patents.

[5] On July 26, 2000, the USPTO published internal guidelines for "Formulating And Communicating Rejections Under 35 U.S.C. 103 For Applications Directed To Computer-Implemented Business Method Inventions" http://www.uspto.gov/web/menu/busmethp/busmeth103rej.htm.

[6] See *Diamond v. Diehr*, 450 U.S. 175, 182, 101 S.Ct. 1048, 67 L.Ed.2d 155 (1981); *Parker v. Flook*, 437 U.S. 584, 98 S.Ct. 2522, 57 L.Ed.2d 451 (1978); *Gottschalk v. Benson*, 409 U.S. 63, 93 S.Ct. 253, 34 L.Ed.2d 273 (1972); and *In re Alappat*, 33 F.3d 1526, 1540-41, 31 USPQ2d 1545, 1554 (Fed. Cir. 1994) (in banc).

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

[8] *Arythmia Research Technology Inc. v. Corazonix Corp*., 958 F.2d 1053

[9] *State Street Bank & Trust Co. v. Signature Financial Group Inc*., 149 F.3d 1368 (Fed. Cir. 1998)

On January 11, 1999, the Supreme Court[10] left intact the decision of the Federal Circuit Court in *State Street Bank*, which held that a computer system designed to implement an investment structure is patentable. The Federal Circuit Court reversed a ruling by the U.S. District Court for the District of Massachusetts[11] holding invalid as unpatentable U.S. Patent No. 5,193,056, entitled "Data Processing System for Hub and Spoke Financial Services Configuration," a system that facilitates the pooling of mutual funds assets in an investment portfolio organized as a partnership. The patent was issued to Signature Financial Group Inc. (Signature) in March 1993. When licensing negotiations broke down between State Street Bank & Trust Co. (State Street) and Signature, State Street brought suit against Signature. The court granted State Street's motion for summary judgment of nonvalidity for failure to claim statutory subject matter under 35 U.S.C. § 101. Signature appealed the decision.

In reversing the District Court's decision, the Federal Circuit stated that while mathematical algorithms are not patentable subject matter to the extent that they are merely abstract ideas, "the transformation of data, representing discrete dollar amounts, by a machine, through a series of mathematical calculations, resulting in a final share price constitutes a practical application of a mathematical algorithm, formula, or calculation, because it produces a 'useful, concrete and tangible result'—a final share price momentarily fixed for recording and reporting purposes and even accepted and relied upon by regulatory authorities in subsequent trades." As a result of the decision, the data processing system "is statutory subject matter, even if the useful result is expressed in numbers, such as price, profit, percentage, cost, or loss."

Essentially, computer software that utilizes a mathematical formula and methods for doing business are now patentable subject matter, which means that we have new and powerful ways to protect these kinds of intellectual property.

Best Mode and Enabling Disclosure

If computer software and business methods are now patentable (having overcome the issue of patentable subject matter), the issues of best mode[12] and providing an enabling disclosure[13] become critical. If a patent is not enabling or fails to disclose the best mode, it may be invalidated, or, at best, be of only minimal value.

A specification that claims an invention requiring implementation through computer software but that fails to set forth the details of the computer programming may present

[10] *Cert. denied* January 11, 1999

[11] *State Street Bank & Trust Co. v. Signature Financial Group Inc.*, 927 F. Supp. 502, 515 (D. Mass 1996)

[12] The "best mode" requirement of 35 U.S.C. § 112 is that the best mode known to the inventor for carrying out the invention be disclosed.

[13] The "enablement" requirement of 35 U.S.C. § 112 is that a patent application describe the invention in sufficient detail so as to enable one skilled in the particular technology to make and use the invention.

questions of whether the experimentation required to write the programming is reasonable or unreasonable.[14]

The court cases generally have followed programming trends in the software industry. In *Northern Telecom, Inc. v. Datapoint Corp.*[15] the Federal Circuit held that the district court erred in holding certain claims of the patent invalid for lack of enablement. The Federal Circuit said that "a programmer of reasonable skill could write a satisfactory program with ordinary effort."[16] The Federal Circuit further explained that:

> A decision on the issue of enablement requires determination of whether a person skilled in the pertinent art, using the knowledge available to such a person and the disclosure in the patent document, could make and use the invention without undue experimentation. It is not fatal if some experimentation is needed, for the patent document is not intended to be a production specification.

When the challenged subject matter is a computer program that implements a claimed device or method, enablement is determined from the viewpoint of a skilled programmer using the knowledge and skill with which such a person is charged. The amount of disclosure that will enable the practice of an invention that utilizes a computer program may vary according to the nature of the invention, the role of the program in carrying it out, and the complexity of the contemplated programming, all from the viewpoint of the skilled programmer. As the court observed in *Sherwood*[17], the writing of a program may require varying degrees of skill:

> In general, writing a computer program may be a task requiring the most sublime of the inventive faculty or it may require only the droning use of clerical skill. The difference between the two extremes lies in the creation of mathematical methodology to bridge the gap between the information one starts with ("the input") and the information that is desired ("the output"). The claimed invention of the '375 patent is not in the details of the program writing, but in the apparatus and method whose patentability is based on the claimed combination of components or steps.

In *Fonar Corp. v. General Electric Co.*[18], the Federal Circuit held that, as a general rule, where software constitutes part of best mode of carrying out the invention, description of such a best mode is satisfied by disclosure of functions of software.[19] The Federal Circuit

---

[14] See *Allen Organ Co. v. Kimball International, Inc.*, 839 F.2d 1556, 1566, 5 USPQ2d 1769, 1777 (Fed. Cir. 1988), *cert. denied*, 488 U.S. 850 (1988).
[15] *Northern Telecom, Inc. v. Datapoint Corp.* 908 F.2d 931, 15 USPQ2d 1321 (Fed. Cir. 1990), *cert. denied*, 498 U.S. 920 (1990)
[16] 908 F.2d at 943, 15 USPQ2d at 1330.
[17] *In re Sherwood*, 204 USPQ 537 (C.C.P.A. 1980), *cert. denied*, 450 U.S. 994 (1981)
[18] *Fonar Corp. v. General Electric Co.*, 107 F.3d 1543
[19] See 107 F.3d at 1549.

clearly stated, “flow charts or source code listings are not a requirement for adequately disclosing the functions of software.”

In *Robotic Vision Systems, Inc. v. View Engineering, Inc.*[20], the Federal Circuit held that, even though the specifications did not indicate how the device moving the sensors was controlled, a software program was the only means capable of carrying out the invention. Therefore, it would have been apparent to one “skilled in the art” to utilize software for the implementation of the scanning method claimed. Furthermore, one “skilled in the art” would have known how to create specific source code for that purpose.[21]

In earlier instances of issued patents that utilized software as part of the invention, source code was originally included in a microfiche appendix, with detailed flow charts and pseudo code also supplied to explain the source code. As the programmability and functionality of software became more sophisticated, the actual source code became less important and eventually was not included in a patent application. Ultimately, a functional description alone appeared to be sufficient[22]. Even the implied use of a computer has been considered sufficient.

Conclusion

This trend leads to the interesting situation in which software can be protected simultaneously by patent, copyright, and trade secret. A patent can protect the functional idea, which is implemented by software, while copyright and trade secret laws can protect the details of the software implementation as an unpublished work[23]. This multifaceted legal protection can provide significant financial and competitive advantages for the software developer, as well as establish barriers.

[20] *Robotic Vision Systems, Inc. v. View Engineering, Inc.*, 112 F.3d 1163
[21] See 112 F.3d at 1165-6
[22] While a functional description may be minimally sufficient, the use of pseudo code and functional flowcharts greatly enhances the readability and understandability of a patent.
[23] Unpublished works containing trade secrets are protected by copyright. However, there are special requirements for submitting documents to the Copyright Office for registration in order to retain trade secret protection.