PROTECTING COMPUTER SOFTWARE: AFTER *APPLE COMPUTER, INC. v. FRANKLIN COMPUTER CORP.*, 714 F.2d 1240 (3d Cir. 1983), DOES COPYRIGHT PROVIDE THE BEST PROTECTION?

By Professor Andrew Beckerman-Rodau
Suffolk University Law School
120 Tremont Street
Boston, MA 02108
arodau@suffolk.edu
www.lawprofessor.org

INTRODUCTION

During the last two decades, the foundation of the United States economy has shifted from primary dependence on labor-intensive, manufacturing industries to dependence on the development of advanced technology. This shift has been especially apparent in the field of computers. The development of computer technology has required, and continues to require, tremendous capital expenditures. As part of the efforts to protect these investments, there has been a resurgence of interest in the field of intellectual property law.

Actions taken by all three branches of the federal government illustrate this increased interest in intellectual property law. First, the United States Supreme Court, which has rarely granted certiorari in patent cases, handed down two significant decisions construing specific sections of the patent law within the last few years.[1] Second, Congress enacted a new copyright act,[2] appointed a commission to study the problems of protecting software,[3] amended the Copyright Act in accordance with the commission's recommendations,[4] amended the patent law,[5] and established the Court of Appeals for the Federal Circuit with exclusive jurisdiction over patent appeals from federal district courts.[6] Third, President Reagan urged Congress to modify the federal antitrust and intellectual property laws to improve the competitiveness of American industries in international markets.[7]

Traditionally, patent law has been invoked to protect technological innovations. While the electronic devices and circuits that comprise the hardware of a computer are proper subject matter for patent protection, the software[8] that is utilized to operate a computer represents subject matter at the periphery of patent law.[9]

The increasing importance of software as a basis for advances in computer technology has necessitated the exploration of other areas of the law as possible sources of protection. Although trade secret law has been the traditional means of protecting software, the wide dissemination of many types of software has greatly hampered the practicability of trade secret protection. As a result, the use of trade secret law is being increasingly displaced by the use of copyright law to protect software. Moreover, in light of the decision by the Court of Appeals for the Third Circuit in Apple Computer, Inc. v. Franklin Computer Corp.,[10] which extended copyright protection to all forms of software,[11] copyright law may provide the best protection currently available. This article will discuss the existing state of patent and trade secret law with regard to computer software, analyze the Apple Computer decision, and assess the 'copyrightability' of software.

I. PATENT LAW AND COMPUTER SOFTWARE

When computer technology was in its infancy, the United States Patent and Trademark Office adhered to the view that all computer programs and program-related inventions were unpatentable subject matter.[12] This view, however, was repudiated by a series of decisions from the United States Supreme Court and the Court of Customs and Patent Appeals, which produced a seemingly simple two-step analysis, applicable on an ad hoc basis, for determining whether patentable subject matter existed.[13]

The two-step analysis, as originally articulated in In re Freeman,[14] required the following: 1) The patent claim[15] must first be evaluated to determine if it directly or indirectly recited an algorithm which is defined as a procedure for solving a given type of mathematical problem; 2) Then, if it was found to relate to an algorithm, the claim was analyzed to determine whether, in its entirety, the claim wholly preempted the use of that algorithm, and was therefore unpatentable subject matter.[16]

The United States Supreme Court modified this approach with respect to the patentability of computer software in Diamond v. Diehr.[17] In Diehr, the Court explicitly held that a computer-related invention was patentable subject matter under the patent law.[18] The invention concerned a method of operating a press to mold and cure rubber.[19] The claimed method constantly measured the temperature in the mold, continually calculated an optimum curing time with a mathematical formula contained in a computer program, and opened the press when the program calculated during to be complete.[20] Justice Rehnquist, writing for the majority, concluded that 'a claim drawn to subject matter otherwise statutory does not become nonstatutory simply because it uses a mathematical formula, computer program or digital computer.'[21] Justice Rehnquist also reiterated the dictum of Chief Justice Burger from an earlier case in which the Chief Justice espoused the broad policy that 'courts should not read into the patent laws limitations and conditions which the legislature has not expressed,'[22] and that the legislative history of the patent law 'informs us that Congress

. . . intended patentable subject matter to 'include anything under the sun that is made by man."[23] Accordingly, the majority opinion in Diehr clearly demonstrates that the Court has taken an expansive view of what is patentable subject matter.

In an attempt to clarify the elements delineating patentable and unpatentable subject matter, the Court in Diehr stated that the claim in question must initially be examined to ascertain whether it contained a mathematical formula.[24] Then it must be determined whether the claim sought patent protection for that formula in the abstract, because such protection was not available for a mathematical formula just as it was not available for a scientific principle or a law of nature.[25] The Court explained that unpatentable subject matter cannot be converted to patentable subject matter merely by limiting the use of the formula to a particular technological environment or by adding insignificant post-solution activity.[26] However, the Court noted, that when a claim utilized a mathematical formula, not otherwise patentable, in a structure or process which, when considered as a whole, was performing a function protectable by patent law, then the claim contained patentable subject matter.[27] The Court concluded that the claims in question protected an industrial process that was within the domain of patentable subject matter, and therefore ruled that the claims were valid.[28]

Therefore, under Diehr, the patentability of a computer-related invention hinges on whether the mathematical formula involved actually interacts with another portion of the claimed invention to transform an article to a different state. If the mathematical formula alone is the embodiment of the claimed invention, then neither a preamble establishing a useful environment, nor a mere post-solution activity performed with the formula, can convert the formula into patentable subject matter.

After the Supreme Court's decision in Diehr, the Court of Customs and Patent Appeals decided four cases during the summer of 1982 that all dealt with the patentability of computer-related inventions: In re Tanner,[29] In re Abele,[30] In re Pardo,[31] and In re Meyer.[32] In these cases, the Court of Customs and Patent Appeals seized the opportunity to refine the two- step analysis that was originally announced in Freeman,[33] and modified in Diehr.

These cases, taken as a group, refined the two-step analysis into its present form.[34] Under the present analysis, the claim in question must first be evaluated to determine whether it clearly includes a mathematical algorithm in the form of a mathematical formula or procedure for solving a mathematical problem; second, if such a mathematical formula or procedure is included, then the claim must be further analyzed to determine whether it includes only the mathematical algorithm and is therefore unpatentable subject matter, or whether it encompasses an application of the algorithm that includes statutory subject matter and is therefore patentable. To prevent undermining the two-step analysis by the inclusion of superfluous statutory subject matter in a claim for an algorithm, the analysis requires the included statutory subject matter to be something other than a field-of-use limitation or a nonessential post- solution activity.

Despite the availability of a succinct test for determining whether computer software or a computer-related invention is patentable subject matter, predictability in separating patentable and unpatentable subject matter is very difficult. A simple legal test is not easily applied to the highly complex area of computer software, especially in view of the relevant sections of the patent law which were enacted prior to the recent explosion of computer technology and without being intended to deal with such technology.[35]

In addition to the difficulty of determining whether computer software comes within the domain of the patent law, other problems exist with utilizing the patent system for the protection of computer software. It is generally expensive to obtain a patent.[36] Also, the lengthy period of time, typically several years,[37] that is required to obtain a patent seriously limits the usefulness of patent protection. Therefore, in the high technology area of computers where the economic life of an innovation may be only a few years, the invention's commercial life may have already expired when a patent is granted. A final problem with patent protection of software is the risk of an infringement suit to enforce the patent, which may result in a judicial determination that the patent is invalid. In such a situation, the owner of the invalid patent is left without protection and the public has complete access to the invention. In view of the aforementioned problems, the patent system, as it presently exists, is clearly not the best way to protect innovative developments made in computer software.

II. MAINTAINING COMPUTER SOFTWARE AS A TRADE SECRET

Trade secret protection is applicable to computer software in practically all jurisdictions[38] and has been the most widely utilized method of obtaining protection for software.[39] Reliance on trade secret protection has resulted from both the questionable applicability of patent and copyright law to software, and from the high level of protection afforded information that qualifies and is maintained as a trade secret.[40]

Although the common law definition of a trade secret varies among the states, the Restatement of Torts provides a generally accepted definition: "A trade secret may consist of any formula, pattern, device or compilation of information which is used in one's business, and which gives him an opportunity to obtain an advantage over competitors who do not know or use it. . . . The subject matter of a trade secret must be secret . . . so that, except by the use of improper means, there would be difficulty in acquiring the information. An exact definition of a trade secret is not possible. Some factors to be considered in determining whether given information is one's trade secret are: (1) the extent to which the information is known outside of his business; (2) the extent to which it is known by employees and others involved in this business; (3) the extent of measures taken by him to guard the secrecy of the information; (4) the value of the information to him and his competitors; (5) the amount of effort or money expended by him in developing the information; (6) the ease or difficulty with which the information could be properly acquired or duplicated by others."[41]

To qualify as a trade secret computer software must satisfy the factors delineated above in the Restatement definition. The software must not be widely known outside of the owner's business and the software must be information that enhances the owner's business.[42] Knowledge of the software should be restricted to employees who require access to the software for legitimate business reasons. Such employees should be subject to a secrecy agreement that legally obligates them not to disclose the software. Further, the owner of the software should take precautions, such as limiting access to the software to only those subject to a secrecy agreement, to prevent other employees and persons outside of the business from gaining access to the software.

In addition to establishing the software as a trade secret, the owner of the software must continuously undertake measures to maintain the secrecy of the software.[43] Employees must be reminded of what is and what is not a trade secret, and the trade secret must be maintained as confidential information for the benefit of the employer. The sale or use of software protected as a trade secret by anyone other than the owner or the owner's employees must be subject to an agreement of confidentiality.[44] Finally, this agreement must clearly indicate that the software is a trade secret that is not to be disclosed to anyone not authorized by the agreement to have access to the software.

A great advantage of trade secret protection for computer software is that the software may be maintained as a secret indefinitely, provided that the requirements of a trade secret are continually satisfied.[45] This allows the software owner to enhance his or her position in a competitive marketplace for as long as the software is maintained as proprietary information.

The virtually unlimited monopoly that trade secret protection provides to the owner of qualified software includes certain inherent risks and problems. If the proprietary software is accidentally or illegally disclosed, all protection is lost.[46] Typically, failure of the trade secret owner to be vigilant in maintaining the software as proprietary information eventually leads to public disclosure. If a trade secret is illegally obtained by industrial espionage, disclosed by a former employee, or disclosed in breach of a confidential agreement, and thereby enters the public domain, the information ceases to be proprietary.[47] In such cases, the party responsible for the illegal act can be held liable to the original trade secret owner, but the rest of the world is free to use the trade secret.[48] Additionally, any person who independently creates the same or similar software is free to use it without the threat of recourse, since trade secret law only provides legal protection against someone who illegally obtains the proprietary information. In other words, if the proprietary software is independently developed and publicly disclosed by someone else, trade secret rights are extinguished.[49]

An additional limitation of utilizing trade secret law to protect computer software results from the difficulty of actually restricting access to proprietary software to authorized users. Most contemporary computer systems are directly accessible via ordinary telephone lines. Unauthorized users can therefore enter and utilize a computer from a remote location through the use of a small personal computer connected to a telephone.[50]

Advances in technology over the past few years have changed computers from large and costly machines to small tabletop machines which are inexpensive enough to be available to a large segment of the population for personal use. The tremendous demand for these personal computers has created a vast and extremely profitable market for software producers. This new market is essentially a consumer market in which a large volume of programs are sold through mail order businesses and retail stores.[51] Trade secret law is an ineffective method of protecting this type of market because trade secret protection requires a limited and strictly controlled use of the protected software.[52]

III. COMBINING TRADE SECRET AND COPYRIGHT LAW TO PROTECT COMPUTER SOFTWARE

Another approach to computer software protection is the concurrent use of trade secret and copyright law. This approach allows copyright law to provide protection in the event that the proprietary software is publicly disclosed. The weight of federal court decisions support the combined use of trade secret law and copyright law to safeguard computer software in appropriate instances.[53] However, at least one federal court has frowned upon such an approach.

The decision in which the court disapproved the combined use of trade secret and copyright law, Videotronics, Inc. v. Bend Electronics,[54] is unclear and misleading and should not present a barrier to concurrent use of the two bodies of law. In Videotronics, the court correctly held that trade secret protection was unavailable for computer software that had been made readily available to the public. In so holding, however, the court noted that once the property interest in the software was covered by the Copyright Act of 1976, relief was unavailable under state trade secret law.[55] Any suggestion implicit in this statement that the Act preempted state trade secret law is erroneous.[56] Such a conclusion is buttressed by the decision in Management Science America, Inc. v. Cyborg Systems, Inc.,[57] in which the court stated that the Act did not preempt state trade secret law.[58] In Management Science, the plaintiff had licensed the use of its computer software package for payroll processing to more than six hundred customers[59] and had permitted potential customers to view the system's documentation. However, both the licensing of the system to a customer and the disclosure to a potential buyer were subject to an explicit written nondisclosure agreement in which the parties acknowledged that the information was proprietary and disclosed in confidence.[60] In order to provide additional protection for the software in the event it was publicly disclosed, the manuals and documentation that comprised the system included statutory copyright notices. In an action against former employees for allegedly utilizing the proprietary information contained in the software, the court concluded that as a matter of law 'the mere act of affixing a copyright notice does not preclude the claim of secrecy' under the Act.[61]

Similarly, the Court of Appeals for the Seventh Circuit has ruled that under the Copyright Act of 1909,[62] the placement of a copyright notice on a computer manual does not automatically preclude trade secret protection.[63] Additionally, in Warrington Associates v. Kellogg Citizens National Bank[64] and Warrington Associates v. Real-Time Engineering Systems,[65] two federal district courts found that the defendants, who breached a nondisclosure agreement by allegedly disclosing a confidential computer system manual which contained a copyright notice, could be liable for both copyright infringement under the 1976 Copyright Act and breach of the agreement under trade secret law.[66]

The Seventh Circuit has recognized that copyright law seeks to protect 'the form of a work,' while trade secret law seeks to protect 'contents or ideas in a work.' Therefore, the Seventh Circuit reasoned, the two forms of protection can exist simultaneously in the same material.[67] In discussing the interaction of the two forms of protection, one federal judge eloquently stated:

"An analysis of the interest secured by copyright and trade secret law makes plain that the claims are not 'equivalent' as intended by the Congress. It is well-settled that copyright protection extends not to an idea itself, but rather to the particular expression used by its author. In contrast, the protection provided by the common law of trade secret misappropriation extends to the very ideas of the

author, subject, of course, to the requirement that the idea has some originality and is as yet undisclosed or disclosed only on the basis of confidentiality. The practical distinction between the two interests is manifest. While disclosure of the expression does not vitiate rights secured by copyright law, that same disclosure may well strip the underlying idea of its confidentiality, and thus its status as a trade secret. To a certain degree the two respective rights in intellectual property interact. To the extent a work has been copyrighted and published, the chances of unprivileged disclosure may increase. But the mere fact that an expression is copyrighted does not, in and of itself, disclose the trade secret or eliminate its mantel of confidentiality."[68]

The essential inquiry in a situation where proprietary information has a copyright notice affixed to it is whether the information qualifies as a trade secret by virtue of the measures taken to maintain it as a secret. If such information is openly and unconditionally sold to the public, only copyright law could be invoked to protect a property interest in the material. Unconditional public sales negate any claim of secrecy and therefore eliminate the use of trade secret law as a means of protection.

Despite any confusion that may exist regarding the coincident use of both trade secret and copyright protection, such use is both proper and recommended in appropriate cases. However, the utility of this approach will most likely be limited to proprietary software which is licensed on a confidential basis to a limited number of customers.[69]

IV. COPYRIGHT PROTECTION FOR SOFTWARE

The limitations of patent and trade secret law have resulted in reliance on copyright law as the major source of protection for software. It is undisputed that the 1976 Copyright Act, as amended by the Computer Software Act of 1980, extends copyright protection to software.[70] There has been dispute, however, over the question whether this protection applies to all forms of software.[71] To clarify the nature of this dispute, a brief overview of the general steps of creating a computer program is necessary.

A. Programming a Computer

After developing a clear understanding of a program's objective, the author or 'programmer' writes a flowchart that illustrates the basic logic of the program. This flowchart utilizes standard symbols and words to lay out the basic logic and is readily understandable by anyone with minimal knowledge of programming. The basic logic is then transferred into a high level computer language such as FORTRAN, BASIC, COBOL, or SNOBOL. These languages consist of english words, abbreviations, numbers, and mathematical symbols. The high level language form of the program is fed into a computer where a special program known as a compiler translates the high level language into assembly language consisting of alphanumeric labels that represent particular steps or operations to be performed. Alternatively, the computer program can be directly written in assembly language and read into the computer.

Programs written in a high level language or an assembly language are known as source programs or source code and are readily understandable by persons skilled in the art of programming. Before a program is understandable to a computer, an assembler program converts the assembly language into machine language, also referred to as object code.[72]

A modern computer works digitally, meaning that it only recognizes combinations of data (called bytes) in which each individual piece of data (referred to as a bit) is represented by one of two states. These states are commonly represented symbolically by a '0' or a '1'. Therefore, the machine language form of the program consists entirely of combinations of bits which, if written out by a programmer, would comprise clusters of '0's' and '1's.' This form of a program can be directly understood only by an expert in machine language who has painstakingly deciphered the combinations of '0's' and '1's.'[73]

All programs, regardless of the form in which they are expressed, can be classified as applications programs or operating programs. The distinction between the two types of programs is based on the function served by the program. Typically, an applications program performs a specific task for the user, such as performing calculations, tabulating data, or producing graphics. In contrast, an operating program is usually an internal computer program which actually runs the computer and the applications programs.

B. Copyrightability of Object Code

There is very little dispute regarding the extension of copyright protection to the flowchart, high level language, or assembly language forms of expressing applications software. However, confusion has existed as to whether machine language, or object code, is a form of expression covered by the Copyright Act.[74] Additionally, the question has arisen whether operating programs in any form are within the domain of the Copyright Act.[75]

The belief that object code is not copyrightable is based in part on the confusion between object code and the medium used to store object code.[76] Object code can be stored on magnetic tape, magnetic discs, or punched cards or in a silicon chip[77] such as a read only memory ('ROM'). The use of ROM's has created the most confusion. A ROM is a small electronic device that produces specific outputs in response to externally supplied data. These outputs result in the execution of certain instructions within the computer.[78] Most of the sections of the ROM are standard and are used in all ROM's. However, the memory section of each ROM has a unique physical structure that is defined by the object code that the ROM embodies.[79] Therefore, the memory section of a ROM can be viewed as a physical device that embodies software in its object code or machine language form.

In Data Cash Systems v. JS&A Group,[80] the Federal District Court for the Northern District of Illinois found that the object code form of a computer program stored in a ROM was not a copy of the program under the 1909 Copyright Act because object code was not in a form that could be seen and read by humans.[81] The court also posited that the Copyright Act of 1976 covers computer programs in flowchart, source, and assembly phases but not in their object code phase.[82] The court viewed the object code form of a program, when stored in a ROM, as a mechanical device that was outside the scope of the copyright law.[83] On appeal, the Court

of Appeals for the Seventh Circuit affirmed the district court's decision without reaching the question whether object code in a ROM was copyrightable.[84] The denial of copyright protection was affirmed simply on the basis of failure to satisfy the notice requirements of the Copyright Act of 1909.[85]

The first appellate decision to consider the question whether a computer program embodied in a ROM could be a copy of a copyrighted program is Williams Electronics, Inc. v. Arctic International, Inc.[86] In that case, the Court of Appeals for the Third Circuit rejected the argument that a copy must be directly understandable to a human being to be considered a copy under the copyright act.[87] In so doing, the court relied upon the wording of the Copyright Act of 1976: 'A copy is defined to include a material object in which a work is fixed by any method now known or later developed, and from which the work can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.'[88] Based on this passage, the court suggested, that the duplication of a copyrighted computer program fixed in a ROM amounted to copyright infringement.[89]

In GCA Corp. v. Chance[90] and Midway Manufacturing Co. v. Strohon,[91] two federal district courts recognized that object code is simply a mechanical transformation of a computer program and is therefore protectable when it is copied from a copyrighted source program.[92] The GCA Court stated: "Plaintiff's source code falls within the protection of copyright laws as a work of authorship fixed in any tangible medium of expression from which it can be perceived, reproduced or otherwise communicated . . .. Because the object code is the encryption of the copyrighted source code, the two are to be treated as one work; therefore copyright of the source code protects the object code."[93]

Although the courts agree that copyright law applies to computer software, they disagree as to the scope of protection. This is especially true with regard to copyright protection for object code.

C. Apple Computer v. Franklin Computer

The first appellate opinion to deal definitively with the copyrightability of software is the Third Circuit's decision in Apple Computer, Inc. v. Franklin Computer Corp.[94] In Apple Computer, the defendant, Franklin, had allegedly copied fourteen of Apple's copyrighted operating programs which were stored on floppy discs and in ROM's.[95] Franklin then allegedly incorporated these duplicated programs into a personal computer on which virtually all software designed for Apple's personal computer could be utilized.[96] Apple brought suit in the United States District Court for the Eastern District of Pennsylvania alleging copyright infringement of its operating programs and seeking a preliminary injunction to prevent the continued use of Apple's operating programs by Franklin. The district court denied the preliminary injunction, in part, because it determined that it was unclear whether the Copyright Act of 1976 applied to operating systems or to object code.[97]

On appeal to the Court of Appeals for the Third Circuit, several issues emerged. The court first considered whether a computer program expressed in object code was copyrightable.[98] The court held that a computer program in source or object code is a literary work under the copyright law, and is therefore protected from unauthorized copying.[99]

In rejecting Franklin's argument that object code is not copyrightable because it is not designed to be read by a human, the court focused on section 102 of the Copyright Act of 1976.[100] The court found that this section defined copyrightable subject matter to include original works of authorship fixed in any tangible medium of expression without regard to whether the subject matter can be directly perceived by someone or understood only through the aid of a machine. Relying on its earlier statements in Williams Electronics and the legislative history of the Copyright Act,[101] the court determined that Congress included the language of section 102 to overrule an earlier United States Supreme Court decision that established that forms of expression not directly perceivable to humans fell outside of the copyright law. 102 Therefore, the fact that object code is not perceivable to humans is irrelevant.

In a single paragraph, the court also relied on Williams Electronics in dismissing Franklin's argument that a computer program embedded in a ROM is not copyrightable because a ROM is a utilitarian object or a machine part.[103] The court viewed the embedding of object codes in a ROM as meeting the statutory requirement of section 102 that copyrightable subject matter be fixed in a tangible medium of expression.[104]

The court also examined the categories of 'works of authorship' enumerated in the copyright law and concluded that the definition of 'literary works' was clearly broad enough to include computer programs in any form.[105] To bolster its conclusion, the court of appeals observed that the Copyright Act explicitly defined computer programs.[106] The court further noted that the Copyright Act set forth a special exception for copying computer programs for certain purposes.[107] Since Congress addressed exceptions for computer programs in the Copyright Act, the court reasoned that computer software is within the scope of the Act.[108]

The court further considered, as a matter of first impression, Franklin's contention that copyright law could not protect operating system software.[109] Franklin distinguished between applications programs, which it conceded were copyrightable, and operating programs, which it argued were not copyrightable.[110] Franklin correctly cited Baker v. Selden[111] and section 102(b),[112] which codified most of Baker, as precluding the extension of copyright protection to ideas, procedures, processes, systems, methods of operation, concepts, principles, or discoveries. However, Franklin's argument that an operating system is per se unprotectable by copyright because it is a process, system, or method of operation was misplaced. Baker and section 102(b) established that copyright law does not protect the underlying idea, but they do not preclude copyright protection for expression of the idea. Thus, the court determined that the underlying idea of the operating programs - which is the method or system employed by the programs for carrying out the particular functions - is protectable, if at all, under patent law rather than under copyright law. However, the court further found that the form of expressing the ideas embodied in operating programs is protectable by copyright law.[113]

The Apple Computer court rejected any distinction between applications and operating programs, reasoning that it was unable to discern a different status for different types of software under the law. The court found support for its view in the legislative history and in the definition of a 'computer program' contained in section 101, which does not distinguish between different types of programs.[114]

Since copyright protection is limited to the form of expression of an underlying idea, the Apple Computer court was also required to establish guidelines to enable the district court on remand to distinguish between the underlying idea and the form of expression of the idea contained in Apple's software. In determining where to separate the underlying idea from the form of expression, the court emphasized that a pragmatic line must be drawn to ensure a balance between competition and protection of expressions of ideas.[115] The court reasoned that since different expressions of the same idea can each be protected by copyright, the idea and the expression of the idea merge when the idea cannot be expressed in different ways. Therefore, when an idea cannot be expressed in different ways, a copyright on the limited means of expression would be tantamount to protection of the idea.[116]

The Apple Computer court announced a test for determining whether software is a protectable form of an expression of an idea, or simply an idea unprotectable under copyright law.[117] Simply stated, the test asks whether other programs can be written or created that perform the same function as the program in question. If other programs are possible, then the program in question is an expression of the idea and therefore copyrightable. However, if other programs are not possible, then copyright protection is unavailable.[118]

The court applied the test, by way of example, to the 'Applesoft' operating program which translates the source code version of programs designed to run on an Apple Computer into object code.[119] The court stressed that the application of the test to this program should focus on the general function of translating source code into object code, rather than the translation of particular source code written for Apple's computer into object code for Apple's computer. Accordingly, the court stated, if other programs could be written or created that translate source code into object code, 'Applesoft' was copyrightable as one form of expression of this translation function. Even if 'Applesoft' represented the only form of expression of an operating program that would allow applications programs designed for Apple's computer to be run on Franklin's computer, 'Applesoft' would still be copyrightable since the underlying idea is the translation function of any source code into object code.[120]

D. Copyright Law

To evaluate the disparity between the cases that limit copyright protection to only certain forms of software[121] and those cases that have found all forms of software copyrightable,[122] it is necessary to carefully examine federal copyright law and its legislative history. The subject matter covered by copyright law is described in section 102 of the Copyright Act. Section 102, which initially requires an original work of authorship,[123] lists seven illustrative categories of authorship.[124] It is clear that the 'literary works' category, which includes works 'expressed in words, numbers, or other verbal or numerical indicia,' encompasses software. Further, the legislative history explicitly states that computer programs are 'literary works' under the 1976 Copyright Act,[125] and that they were probably covered under the previous Copyright Act.[126]

In addition to the passage of the Copyright Act of 1976, Congress established the National Commission on New Technological Uses of Copyrighted Works ('CONTU').[127] The purpose of CONTU was to study and make recommendations for changes in the portions of the copyright law that concerned the reproduction and use of copyrighted works and the creation of new works with automatic systems capable of storing, processing, retrieving, and transferring information. The Final Report[128] of CONTU concluded that Congress intended to include computer programs as copyrightable subject matter under the Copyright Act of 1976.[129] The Report also recommended the continuation of protection for computer programs and the amendment of the Copyright Act of 1976 to clarify the intent to provide such protection.

Upon completion of CONTU's Final Report, Congress adopted the majority portion of the report[130] and passed the Computer Software Act of 1980.[131] This Act amended the copyright law by adding a definition of 'computer program'[132] to section 101. Additionally, section 117 of the Copyright Act was completely rewritten to establish a special exception under which reproductions and adaptations of computer programs do not constitute infringement under certain circumstances.[133] Although the Copyright Act does not explicitly state that software is copyrightable, the existence of this exception makes it clear by negative implication, that the Act, as amended, extends protection to computer programs as works of authorship.

Once it has been initially established that a computer program is an original work of authorship, section 102 requires the work to be fixed in a tangible medium of expression. The operative words 'fixed in a tangible medium of expression' are defined to include any material object that permits the work of authorship to be perceived or communicated with or without the aid of a machine or device for more than a transitory duration.[134] Under this definition, an original computer program written on paper or contained on computer punch cards[135] would be copyrightable because it is fixed in a tangible medium. It therefore can be argued that object code embedded in a ROM is copyrightable because the ROM represents a permanent and tangible medium of expression for the embedded program.[136] At least one court, however, has said that a computer program can only be an original work of authorship fixed in a tangible medium of expression where the program is expressed as a flowchart, or in source code or assembly language.[137] Thus, according to the court, a computer program in object code is not copyrightable.[138]

This distinction between various equivalent forms of expression of a program is inconsistent with section 106, which grants to the copyright owner the exclusive right to make copies of, and prepare derivative works from, the protected work.[139] Section 106, read in conjunction with the relevant definitions in section 101, clarifies that the right to make copies 'means the right to produce a material object in which the work is duplicated, transcribed, imitated, or simulated in a fixed form from which it can be 'perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.'"[140] Consequently, once copyrightable subject matter is created and fixed in a tangible medium of expression such as a book, computer punch cards, or microfilm, the

material cannot be freely used simply by converting it to an equivalent form that is merely a copy of the original work of authorship.[141]

Another argument that has been asserted against copyright protection for certain programs, such as operating programs, is that they are a process, system or method of operation that is barred from copyright protection by section 102(b).[142] This argument represents a misunderstanding of a fundamental limitation of the copyright law. Section 102(b) merely restates the general principle of law that copyright protection does not extend to the underlying idea of the work of authorship.[143] In the context of computer programs, this principle means that the expression utilized by the programmer is the copyrightable aspect of the program and that the actual processes or methods employed by the program are not covered by the copyright law.[144] The fact that a computer program represents or embodies a method is irrelevant because only the form of expressing the method and not the method itself is within the domain of the copyright law.

The argument that a ROM is really a machine part, or a utilitarian object, and therefore not copyrightable has also been raised.[145] This view represents an additional misunderstanding of the copyright law. The classification of a ROM as a machine part or utilitarian object is of no importance because a ROM is merely a device that embodies copyrightable subject matter. Copyright protection does not extend to, or restrict the use of, ROM's any more than a valid copyright on a book restricts the use of books.[146]

At some point, at least theoretically, the idea and expression of the idea may merge so that a copyright on the expression protects the underlying idea. [147] One leading authority, however, believes that such a merger is not actually possible, reasoning that no system or method can be performed with a single form of expression.[148] This reasoning is especially applicable to software, which can take many different forms and still produce the same result.[149]

The potential merger of the form of expression and the underlying idea is one of the reasons for the distinction that has been made between applications and operating programs in discussions regarding copyright protection. It has been argued that the number of ways available to express an operating program for a particular computer is very limited, and that therefore a copyright on a particular form of expressing an operating program monopolizes the underlying idea of the program.[150] The purpose or underlying idea of an operating program is to control a microprocessor,[151] which in turn operates a computer. However, different operating programs can be used to control the various microprocessors that are available today,[152] and therefore a copyright on an operating program does not monopolize the underlying idea of the program.

The Apple Computer court recognized the theoretical possibility of a merger of the expression and the idea when it declined to find operating programs per se copyrightable. Instead, the Apple Computer court took the approach that operating programs would be copyrightable unless the underlying idea of the program could not be expressed in more than one way.[153] While allowing an operating program to be copyrightable, this approach recognizes that it is theoretically possible for the form of expression to merge into the uncopyrightable idea.

The copyright law therefore provides protection for all forms of software that are original works of authorship manifested in a tangible medium of expression. This protection is available for programs expressed in source code or object code. Furthermore, it extends to both applications and operating programs even where they are contained in a ROM. Copyright protection, however, never extends to the underlying idea embodied in the copyrighted software, or to the medium or device used to express the software.

E. Policy Considerations

The tremendous importance of computer technology as a major part of our economy makes it imperative that society encourage the development and disclosure of technological advances--including the creation of new software. One method of encouraging development and subsequent public disclosure is to grant some form of limited protection to software originators.[154] In this way, the programmer can reap commercial benefits in return for increasing the public's knowledge.[155] In order to effect this underlying policy of securing knowledge for the public, copyright protection should cover all forms of software--including operating programs and object code in a ROM.[156] Failure to extend copyright protection to operating programs or object code stored in a ROM would render copyright protection for software worthless because the public would be able to freely copy and utilize operating and applications programs in their object code form without reserving any valuable or useful rights to the software creator. [157]

Copyright law provides the best currently available means of encouraging software development and public disclosure because such protection is easily obtained at a minimal cost. Once software, which is an original work of authorship, is fixed in a tangible medium,[158] copyright protection exists automatically.[159] The copyright owner must then simply display a proper copyright notice on the tangible medium to notify the public of the copyright.[160] Although the law requires payment of a small fee and the deposit of the copyrighted material in the Library of Congress in order to register a copyright, failure to perfect registration will not affect rights in the copyrighted material.[161]

The minimal expense and speed with which copyright protection is attained is very advantageous. First, the avoidance of an administrative proceeding to obtain rights in software prevents waste of a significant portion of the short commercial life of software.[162] Copyright law also allows numerous independent programmers, who would be unable to afford costly forms of protection, such as provided by the patent law, to obtain protection.[163]

The ease with which copyright protection is obtained is accompanied by a corresponding reduction in the extent of protection. Trade secret and patent protection, which grant absolute monopolies,[164] require significant expenditures of time and effort. Rather than an absolute monopoly, copyright law merely protects the form of expression of software but not the idea or method underlying the computer program. Once copyrighted software is made available to the public, anyone is free to make use of the idea that the software embodies. Therefore, the copyright law provides easily obtainable and sufficiently valuable protection to encourage software development and concurrently limits the extent of the copyright owner's protection in order to insure that the public gains

significant freedom to utilize the teachings of the software.

CONCLUSION

The tremendous growth of computer technology and the resulting need for software has magnified the importance of intellectual property law for protecting computer software.[165] Patent law provides a questionable means of protecting software because the procedure for obtaining a patent is expensive and time-consuming, and the applicability of patent law to software remains unclear. While trade secrecy was once the chief means of protecting software, the mass marketing of multiple copies of computer programs--especially for the personal computer market--has rendered trade secret law an ineffective mechanism for protecting most software. The Copyright Act of 1976, as amended by the Computer Software Act of 1980, is now the chief vehicle for protecting software.

Although the importance of protecting software is undisputed[166] and despite the applicability of copyright law to software,[167] judicial application of copyright law has been both ambiguous and conflicting.[168] Some courts have found copyright protection available for software only where such software is embodied in a particular form and in a particular tangible medium.[169] As a result, a program that has been written in source code will be protected by copyright, while the translation of that program to object code will not enjoy copyright protection. Confusion between the protected form of expression and the medium of expression has resulted in the argument that a protected program can be freely used once it is embedded in a ROM. Furthermore, confusion between the form of expression and the underlying idea of the program allowed the novel argument to be raised in Apple Computer that applications software, but not operating software, was within the domain of copyright law. [170]

The judicial distinctions among types of software, forms of expression of software, and different mediums containing the software were eliminated by the Apple Computer court. The court categorically affirmed that software is protectable under copyright law. The only limitation that the court expressed was that operating system software would not be protectable if it could be shown that the function of the operating software could be accomplished by only one form of expression of the software.[171]

The Apple Computer decision represents both the majority view and the correct view of the treatment of software under copyright law. It therefore seems clear that the copyright law will be the major source of protection of software once the extent of protection is increased by the elimination of artificially imposed distinctions between the types of software and their forms of expression.

FOOTNOTES

1. See Diamond v. Diehr, 450 U.S. 175, 187 (1981) (process for curing rubber that included use of a computer held patentable); Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980) (live, human-made micro-organism held patentable).

2. The Copyright Act of 1976, Pub. L. No. 94-553, 90 Stat. 2541, (codified at 17 U.S.C. sections 101-702 (1982)). The Act became effective on January 1, 1978.

3. National Commission on New Technological Uses of Copyrighted Works, Pub. L. No. 93-573, 88 Stat. 1873 (1974) (codified at 17 U.S.C. section 701 note (1982)) hereinafter cited as CONTU. CONTU published a final report which is reproduced in its original form in Final Report of the National Commission on New Technological Uses of Copyrighted Works, 3 COMPUTER L.J. 53 (1981-1982) hereinafter cited as Final Report.

4. Computer Software Act, Pub. L. No. 96-517, sections 10(a)-10(b), 94 Stat. 3015, 3028 (1980) (codified as amended at 17 U.S.C. sections 101, 117 (1982)).

5. See, e.g., Pub. L. No. 97-247, 96 Stat. 317 (1982) (codified in scattered sections of 35 U.S.C.); Pub. L. No. 96-517, 94 Stat. 3015 (1980) (codified in scattered sections of 35 U.S.C.). See also 37 C.F.R. sections 1.1-1.570 (1983) (revised regulations of the Patent and Trademark Office).

6. See Federal Courts Improvement Act of 1982, s 101, Pub. L. No. 97-164, 96 Stat. 25 (1982) (codified at 28 U.S.C. section 1 note (1982)). The United States Court of Appeals for the Federal Circuit replaced the United States Court of Customs and Patent Appeals and assumed appellate jurisdiction of patent matters.

7. See Legislation: President proposes antitrust reforms in attempt to promote innovation, 26 PAT. TRADEMARK & COPYRIGHT J. (BNA) No. 646, at 445 (Sept. 15, 1983); see also National Productivity and Innovation Act of 1983, id. at 456-62 (President Reagan's proposed modification of the antitrust and intellectual property laws).

8. 'A 'computer program' also known as software is a set of statements or instructions used directly or indirectly in a computer in order to bring about a certain result.' 17 U.S.C. section 101 (1982).

9. The patent law limits patentable subject matter to 'any new and useful process, machine, manufacture or composition of matter, or any new and useful improvement thereof.' 35 U.S.C. section 101 (1982).

10. 714 F.2d 1240 (3d Cir. 1983), cert. denied, 104 S.Ct. 690 (1984).

11. The Third Circuit specifically held that the copyright statute provided protection for computer programs. Id. at 1248. The parties subsequently reached an out-of-court settlement under which Franklin agreed to pay Apple 2.5 million dollars and to install its own operating systems in Franklin computers. Copyright, Settlement, 27 PAT. TRADEMARK & COPYRIGHT J. (BNA) NO. 662, at 228 (Jan. 12, 1984).

12. See In re Freeman, 573 F.2d 1237, 1244 (C.C.P.A. 1978) (Patent and Trademark Office erroneously concluded that system of typesetting alphanumeric information with computer-based control system was unpatentable).

13. The following cases should be examined to understand the evolution of the test: In re Freeman, 573 F.2d 1237, 1245 (C.C.P.A. 1978) (first statement of two-step analysis); In re Bradley, 600 F.2d 807, 813 (C.C.P.A. 1979) (expressly applied Freeman two-step analysis), aff'd. per curiam by an equally divided Court sub nom., Diamond v. Bradley, 450 U.S. 381 (1981) (mem.); Diamond v. Diehr, 450 U.S. 175, 187 (1981) (computer-related invention patentable provided protection not being sought solely for mathematical formula alone); In re Taner, 681 F.2d 787, 790 (C.C.P.A. 1982) (method that relies on algorithm is patentable provided it does not preempt use of algorithm); In re Abele, 684 F.2d 902, 907 (C.C.P.A. 1982) (second step of analysis requires claim to be viewed without mathematical algorithm to see if patentable subject matter remains); In re Pardo, 684 F.2d 912, 915 (C.C.P.A. 1982) (use of word 'algorithm' merely to describe process in question is insufficient basis for holding that process is not statutorily protected); In re Meyer, 688 F.2d 789, 795-96 (C.C.P.A. 1982) (mathematical algorithm indicates presence of scientific principle, law of nature, idea, or mental process which are unpatenable if not applied to patentable subject matter).

14. 573 F.2d at 1245.

15. A 'claim' is a required statement at the end of a patent that specifically defines the metes and bounds of the invention that is granted patent protection. 35 U.S.C. section 112 (1982).

16. 573 F.2d at 1245.

17. 450 U.S. 175 (1981). See supra note 12 and accompanying text for a discussion of other Supreme Court decisions addressing the patentability of computer software.

18. 450 U.S. at 187.

19. Id. at 177.

20. Id. at 187.

21. Id.

22. Id. at 182 (quoting Diamond v. Chakrabarty, 447 U.S. 303, 308 (1980)).

23. 450 U.S. at 182 (quoting Diamond v. Chakrabarty, 447 U.S. 303, 309 (1980)).

24. 450 U.S. at 119.

25. Id.

26. Id. at 191-92.

27. Id. at 192.

28. Id. at 192-93.

29. 681 F.2d 787 (C.C.P.A. 1982). 30. 684 F.2d 902 (C.C.P.A. 1982).

31. 684 F.2d 912 (C.C.P.A. 1982).

32. 688 F.2d 789 (C.C.P.A. 1982). For further discussion of these four decisions, see supra note 13.

33. See supra notes 13-15 and accompanying text for discussion of Freeman and the two-step analysis.

34. Now that the Court of Appeals for the Federal Circuit, which is the successor of the Court of Customs and Patent Appeals, has been granted exclusive jurisdiction over patent appeals, see supra note 6, the above two- step analysis for determining patentability of computer-related inventions represents the current state of the law in all the federal circuits. See, e.g., South Corp. v. United States, 690 F.2d 1368, 1371 (Fed. Cir. 1982) (Court of Appeals for the Federal Circuit adopted all prior decisions of Court of Customs and Patent Appeal's as binding precedent on new court).

35. See 35 U.S.C. sections 101-103 (1982) (establishing requirements for patentability).

36. See Bender, Computer Software Licensing, 1983 PROTECTING TRADE SECRETS 405, 424 n.1, in which the author notes that

the cost of obtaining a patent can be as high as $100,000.

37. The interim between filing a patent application and the issuance of a patent is almost 3 years. Id. at 424. In the area of 'Electronic Component Systems and Devices,' the actual filing date of the oldest new patent application awaiting action is August 25, 1980. See 1035 Off.Gaz.Pat. Office 7 (1983). Despite modernization of the Patent and Trademark Office and resulting increases in efficiency, Donald J. Quigg, Deputy Commissioner of Patents and Trademarks, has indicated that the goal of the office is to achieve, by 1987, an 18-month pendency period for applications. Conference Reviews; PTO Rule Changes, 26 PAT. TRADEMARK & COPYRIGHT J. (BNA) NO. 649, at 507 (Oct. 6, 1983).

38. Cybertek Computer Prods., Inc. v. Whitfield, 203 U.S.P.Q. 1020 (Cal. Super. Ct. 1977). See generally J & K Computer Syss., Inc. v. Parrish, 642 P.2d 732, 735 (Utah 1982) (trade secret protection is appropriate for computer programs intended to be kept confidential).

39. 1 R. MILGRIM, TRADE SECRETS section 2.06A5c, at 2-72.20 (1984); Bender, Trade Secret Protection of Software, 38 GEO.WASH.L.REV. 909 (1969-1970).

40. Although trade secret law, which consists of state common law augmented by state statutes, is the antithesis of the patent law, the Supreme Court has held that state trade secret law is not preempted by the patent law. Kewanee Oil Co. v. Bicron, 416 U.S. 470, 484 (1974) (trade secret law and patent law can never conflict).

41. RESTATEMENT OF TORTS section 757 comment b (1939). See R. Milgrim, supra note 35, section 2.01, at 2 n.2 for a detailed listing of cases relying on the Restatement definition. See also Forest Labs., Inc. v. Pillsbury Co., 452 F.2d 621, 624 (7th Cir. 1971) and Abbott Labs. v. Norse Chemical Corp., 33 Wis.2d 445, 463-64, 147 N.W.2d 529, 538-39 (1967), in which the courts stated that the factors listed in the Restatement definition were to be examined when determining whether information was a trade secret.

42. Grogan, Decompilation and Disassembly: Undoing Software Protection, 1 THE COMPUTER LAW. 1, 2 (Feb. 1984).

43. Gilburne, The Proprietary Rights Pyramid: An Integrated Approach to Copyright and Trade Secret Protection for Software, 1 THE COMPUTER LAW. 1, 3 (March 1984).

44. Id.

45. See Aktiebolaget Bofors v. United States, 194 F.2d 145, 147 (D.C. Cir. 1951) (property right in unpatented trade secret exists as long as owner does not disclose secret).

46. D. REMER, LEGAL CARE FOR YOUR SOFTWARE 20 (1982).

47. Id. at 143.

48. Id. See also E. I. du Pont de Nemours & Co. v. Christopher, 431 F.2d 1012, 1016 (5th Cir. 1970) (one who discloses or uses another's trade secret obtained improperly or in breach of confidence is liable in tort to trade secret owner).

49. D. REMER, supra note 46, at 20. But cf. 35 U.S.C. sections 154, 271 (1982) (patent prevents anyone from utilizing, manufacturing, or selling protected invention for 17 years, regardless of whether the invention was copied or independently created).

50. See Beware: Hackers at Play, NEWSWEEK, Sept. 5, 1983, at 42-48.

51. Grogan, supra note 42, at 3. See also Note, Copyright Protection of Computer Program Object Code, 96 HARV.L.REV. 1723, 1723 (1983) (programs now widely distributed over-the-counter to consumers as result of ubiquity of personal computers).

52. Final Report, supra note 3, at 69-70 (trade secret protection dissolves upon disclosure). Miles R. Gilburne stated at the American Patent Law Association meeting held in Palm Springs, California in February 1984 that, because of the explosion of software distribution channels, copyright law and not trade secret law was currently the appropriate way to protect software. American Pat. Law Ass'n.; 1984 Mid-Winter Conf. Summary, 27 PAT. TRADEMARK & COPYRIGHT J. (BNA) No. 669, at 421 (March, 1, 1984). Although it is conceivable that as a condition of sale all retail purchasers would agree to keep the contents of software secret, and thereby preserve trade secret protection, it is unlikely that the mechanics of such a scheme could work. It seems improbable that a court would allow the law of trade secrets to apply in a case where there was an attempt to simultaneously maintain information as proprietary, while also disclosing it to the public. See Videotronics, Inc. v. Bend Elecs., 564 F.Supp. 1471, 1475 (D. Nev. 1983) (public disclosure of trade secret destroys basis of trade secret protection). But see Management Science Am., Inc. v. Cyborg Sys., Inc., 6 Computer L.Serv.Rep. (Callaghan) 921, 922 n.1 (N.D. Ill. 1978) (disclosure of software, subject to a confidential agreement, to more than 1,300 customers did not destroy proprietary nature of the software).

53. See, e.g., Technicon Medical Information Syss. v. Green Bay Packaging, Inc., 687 F.2d 1032 (7th Cir. 1982); Warrington Assocs. v. Real-Time Eng. Syss., 522 F.Supp. 367 (N.D. Ill. 1981); Warrington Assocs. v. Kellogg Citizens Nat'l Bank, 215 U.S.P.Q. (BNA) 375 (E.D. Wis. 1981); Management Science Am., Inc. v. Cyborg Syss., Inc., 6 Computer L.Serv.Rep. (Callaghan) 921 (N.D. Ill. 1978). But see Videotronics, Inc. v. Bend Elecs.; 564 F.Supp. 1471, 1476 (D. Nev. 1983) (court suggests copyright law preempted state trade secret law with regard to computer programs).

54. 564 F.Supp. 1471 (D. Nev. 1983).

55. Id. at 1477.

56. The suggestion that federal copyright law preempted state trade secret law may have been based on the court's failure to differentiate between a property interest in the form of expression (which can be protected by copyright), and a property interest in the underlying idea (which can be protected by patent or trade secret law). It is possible that the court's statement may have referred only to protection for the form of expression, since the court had already found state trade secret protection unavailable. Id. at 1475-76. Read in this light, the court's conclusion is correct because state law cannot protect the form of expression when the federal copyright law is applicable. However, in the absence of such an interpretation, the statement reveals the court's misconstruction of the relationship between the two bodies of law. The Copyright Act of 1976 clearly preempts and abolishes all common law equivalents to rights covered by the Act and applicable to works within the scope of the Act. 17 U.S.C. section 301 (1982). The purpose of section 301 was to eliminate the previous system under the Copyright Act of 1909. Under that system, the common law protected work prior to its publication and the Copyright Act of 1909 protected the work subsequent to publication. See H.R. REP. NO. 1476, 94th Cong., 2d Sess. 129, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5745 hereinafter cited as H.R. REP.. However, under the Copyright Act of 1976, copyright protection arises automatically in a covered work once it is fixed in a tangible medium. Id. Section 301(b) of the 1976 Act makes it clear that states are free to control rights that are not within the domain of, or equivalent to, rights under federal copyright law. Therefore, state statutes and the common law, which both protect ideas, are not in conflict with federal copyright law because federal copyright protection is limited to the form of expression of an idea, rather than the underlying idea itself. See H.R. REP., supra, at 130-31, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5745-47. See also Mazer v. Stein, 347 U.S. 201, 217 (1954); Eden Toys, Inc. v. Marshall Field & Co., 675 F.2d 498, 500 (2d Cir. 1982); Miller v. Universal City Studios, Inc., 650 F.2d 1365, 1368 & n.1 (5th Cir. 1981); Gero v. Seven-Up Co., 535 F.Supp. 212, 216 (E.D.N.Y.), aff'd, 714 F.2d 113 (2d Cir. 1982); Atari, Inc. v. Amusements World, Inc., 547 F.Supp. 222, 228 (D. Md. 1981); Midway Mfg. Co. v. Dirkschneider, 543 F.Supp. 466, 480 (D. Neb. 1981); Dr. Pepper Co. v. Sambo's Restaurants, Inc., 517 F.Supp. 1202, 1207 (N.D. Tex. 1981). It should be noted that state trade secret law is actually equivalent to the federal patent law. Warrington Assoc. v. Real-Time Eng. Syss., Inc., 522 F.Supp. 367, 369 (N.D. Ill. 1981) (trade secret law and federal patent law are congruent). Therefore, it can be more logically argued that the patent law, rather than the copyright law, preempted state trade secret law. It is well-established, however, that patent and trade secret law coexists. Kewanee Oil Co. v. Bicron Corp., 416 U.S. 470 (1974); Pitney Bowes, Inc. v. Mestre, 701 F.2d 1365, 1371 (11th Cir.), cert. denied, 104 S.Ct. 239 (1983).

57. 6 Computer L.Serv.Rep. (Callaghan) 921 (N.D. Ill. 1978).

58. Id. at 926 n.4.

59. Id. at 922. Furthermore, a subsystem of the software package was licensed to more than 1,300 customers. Id.

60. Id. at 922-23.

61. Id. at 924.

62. 17 U.S.C. sections 101-810 (1982).

63. Technicon Medical Syss., Inc. v. Green Bay Packaging, Inc., 687 F.2d 1032, 1038 (7th Cir. 1982), cert. denied, 103 S.Ct. 732 (1983).

64. 1981-1983 COPYRIGHT L. DEC. (CCH) P25,345 (E.D. Wis. 1981).

65. 522 F.Supp. 367 (N.D. Ill. 1981).

66. Kellogg, 1981-1983 Copyright L. DEC. (CCH) P25,345, at 16,941; Real Time, 522 F.Supp. at 368.

67. Technicon Medical Syss., Inc. v. Green Bay Packaging, Inc., 687 F.2d 1032, 1038 (7th Cir. 1982), cert. denied, 103 S.Ct. 732 (1983).

68. Warrington Assoc. v. Real-Time Eng. Syss., 522 F.Supp. 367, 368 (N.D. Ill. 1981) (citations omitted).

69. See, e.g., Management Science Am., Inc. v. Cyborg Syss., Inc., 6 Computer L.Serv.Rep. (Callaghan) 921 (N.D. Ill. 1978) (limited number of proprietary software manuals licensed under a confidential nondisclosure agreement also had copyright notices affixed to them).

70. See, e.g., Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983); Williams Elecs., Inc. v. Arctic Int'l Inc., 685 F.2d 870 (3d Cir. 1982); Hubco Data Prods. v. Management Assistance, Inc., 219 U.S.P.Q. (BNA) 450 (D. Idaho 1983); Midway Mfg. Co. v. Strohon, 564 F.Supp. 741 (N.D. Ill.), aff'd, 704 F.2d 1009 (7th Cir.), cert. denied, 104 S.Ct. 90 (1983); Apple Computer, Inc. v. Formula Int'l, Inc., 562 F.Supp. 775 (C.D. Cal. 1983), aff'd, 725 F.2d 521 (9th Cir. 1984); Midway Mfg. Co. v. Arctic Int'l, Inc., 547 F.Supp. 999 (N.D. Ill. 1982), aff'd, 704 F.2d 1009 (7th Cir. 1983); GCA Corp. v. Chance, 217 U.S.P.Q. (BNA) 718 (N.D. Cal. 1982); Tandy Corp. v. Personal Micro Computers, Inc., 524 F.Supp. 171 (N.D. Cal. 1981); Data Cash Syss., Inc. v. JS&A Group, Inc., 480 F.Supp. 1063 (N.D. Ill. 1979), aff'd, 628 F.2d 1038 (7th Cir. 1980); Synercom Technology, Inc. v. University Computing Co., 462 F.Supp. 1003 (N.D. Tex. 1978). See also Final Report, supra note 3, at 54 (recommendation to extend copyright protection to software).

71. See Data Cash Syss., Inc. v. JS&A Group, Inc., 480 F.Supp. 1063, 1067- 68 n.4 (N.D. Ill. 1979) ('the 1976 Act applies to

computer programs in their flowchart, source, and assembly phases but not in their object phase'), aff'd, 628 F.2d 1038 (7th Cir. 1980); In re Data Gen. Corp. Antitrust Litig., 529 F.Supp. 801, 816 (N.D. Cal. 1981) (copyright law does not necessarily prevent others from copying material embodiment of a program); In re Data Gen. Corp. Antitrust Litig., 490 F.Supp. 1089, 1113 (N.D. Cal. 1980) (copyright notices on software protect underlying source program but not object program). Compare Stern, Another Look at Copyright Protection of Software: Did the 1980 Act Do Anything for Object Code?, 3 COMPUTER L.J. 1, 1 (1981) (object code not copyrightable) with Note, Copyright Protection of Computer Program Object Code, 96 HARV.L.REV. 1723, 1732 (1983) (object code copyrightable).

72. E.g., Apple Computer, Inc. v. Franklin Computer Corp., 545 F.Supp. 812, 821, 822 (E.D. Pa. 1982) (programs are sometimes written directly in object code), rev'd on other grounds, 714 F.2d 1240 (3d Cir. 1983), cert. denied, 104 S.Ct. 690 (1984).

73. Prior to the invention of source languages, programmers had to write directly in object code. Schmidt, Legal Proprietary Interest in Computer Programs: The American Experience, 21 JURIMETRICS J. 345, 352 (1981).

74. See supra note 71 for a survey of the case law and commentary addressing the copyrightability of object code.

75. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1252 (3d Cir. 1983).

76. Section 202 of the 1976 Copyright Act distinguishes between copyright in the work of authorship and the material object which embodies the work. As pointed out by the Act's legislative history: 'The principle restated in section 202 is a fundamental and important one: that copyright ownership and ownership of a material object in which the copyrighted work is embodied are entirely separate things.' H.R. REP., supra note 56, at 124, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5739.

77. See Final Report, supra note 3, at 87. 78. See Williams Elecs., Inc. v. Arctic Int'l, Inc., 685 F.2d 870, 872 (3d Cir. 1982); Apple Computer, Inc. v. Formula Int'l, Inc., 562 F.Supp. 775, 778 (C.D. Cal. 1983), aff'd, 725 F.2d 521 (9th Cir. 1984).

79. Apple Computer, Inc. v. Franklin Computer Corp., 545 F.Supp. 812, 813 (E.D. Pa. 1982), aff'd, 714 F.2d 1240 (3d Cir. 1983), cert. denied, 104 S.Ct. 690 (1984); G. McWHORTER, UNDERSTANDING DIGITAL ELECTRONICS G-9 (1978).

80. 480 F.Supp. 1063 (N.D. Ill. 1979), aff'd, 628 F.2d 1038 (7th Cir. 1980).

81. 480 F.Supp. at 1069. 82. Id. at 1067 n.4. See also In re Data Gen. Antitrust Litig., 529 F.Supp. 801, 816 (N.D. Cal. 1981), which involved a series of antitrust actions challenging the legality of tying licensing of software to the sale of computer hardware, whereby the court suggested that a copyright notice on software protects the underlying source code but not the object code.

83. 480 F.Supp. at 1069.

84. Data Cash Sys. v. JS&A Group, Inc., 628 F.2d 1038 (7th Cir. 1980).

85. 17 U.S.C. section 101-810 (1982).

86. 685 F.2d 870 (3d Cir. 1982).

87. Id. at 877.

88. Id. (quoting 17 U.S.C. s 101 (1976)).

89. 685 F.2d at 877.

90. 217 U.S.P.Q. (BNA) 718 (N.D. Cal. 1982).

91. 564 F.Supp. 741 (N.D. Ill. 1983).

92. Id. at 750-51; GCA Corp., 217 U.S.P.Q. (BNA) at 720.

93. 217 U.S.P.Q. (BNA) at 720 (citations omitted).

94. 714 F.2d 1240 (3d Cir. 1983). See supra note 10 for a brief account of the subsequent history of Apple Computer.

95. See Apple Computer, 545 F.Supp. 812, 813, 815 (E.D. Pa. 1982).

96. Id. at 814-15.

97. Id. at 825.

98. 714 F.2d at 1246.

99. Id. at 1249; The Apple Computer court carefully examined sections 101 and 102 of the Copyright Act of 1976, section 117 which was amended in 1980, the legislative history of the Act, and the final report of CONTU before reaching its conclusion. Id. at 1247-49. The court also reviewed its earlier decision in Williams Elecs. in which the Third Circuit found software to be copyrightable under the Act as amended in 1980. Id. at 1247. 100. 17 U.S.C. section 102 (1982).

101. H.R. REP., supra note 56, at 129, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5759.

102. 714 F.2d at 1248 (section 102 overruled White-Smith Music Publishing Co. v. Apollo Co., 209 U.S. 1 (1908)).

103. 714 F.2d at 1249.

104. Id.

105. Id. at 1248.

106. Id. at 1247-48 (citing 17 U.S.C. section 101 (1982)).

107. 714 F.2d at 1248 (citing 17 U.S.C. section 117 (1982)).

108. 714 F.2d at 1249.

109. Id. at 1250. Although the Apple Computer court was the first United States court of appeals to consider a distinction between operating and applications programs, at least one district court had already considered this question. In Apple Computer, Inc. v. Formula Int'l Inc., 562 F.Supp. 775, 780 (N.D. Cal. 1983), aff'd, 725 F.2d 521 (9th Cir. 1984), the district court found that all types of software were copyrightable, and that no distinction existed between operating programs that controlled computer operations and other types of programs. See also Tandy Corp. v. Personal Micro Computers, Inc., 524 F.Supp. 171, 174-75 (N.D. Cal. 1981) (operating program in a ROM that translated input information from computer operator into machine language was within scope of 1976 Copyright Act).

110. 714 F.2d at 1249.

111. 101 U.S. 99 (1879).

112. 17 U.S.C. section 102(b) (1982).

113. 714 F.2d at 1252.

114. Id. 115. Id. at 1253.

116. Id. 117. Id.

118. Id.

119. See id. See also Apple Computer, 545 F.Supp. at 815.

120. 714 F.2d at 1253.

121. See supra note 65 for a survey of the cases that limit copyright protection to certain forms of software.

122. See Apple Computer, 714 F.2d at 1249; Apple Computer, Inc. v. Formula Int'l, Inc., 562 F.Supp. 775, 783 (C.D. Cal. 1983), aff'd, 725 F.2d 521 (9th Cir. 1984). See also Hubco Data Prods. v. Management Assistance Inc., 219 U.S.P.Q. (BNA) 450 (D. Idaho 1983); Midway Mfg. Co. v. Strohon, 564 F.Supp. 741, 750 (N.D. Ill.), cert. denied, 104 S.Ct. 90 (1983); Midway Mfg. Co. v. Arctic Int'l, Inc., 547 F.Supp. 999, 1012 (N.D. Ill. 1982), aff'd, 704 F.2d 1009 (7th Cir. 1983); GCA Corp. v. Chance, 217 U.S.P.Q. (BNA) 718, 720 (N.D. Cal. 1982); Tandy Corp. v. Personal Micro Computers, Inc., 524 F.Supp. 171, 173 (N.D. Cal. 1981); Synercom Technology, Inc. v. University Computing Co., 462 F.Supp. 1003, 1015 (N.D. Tex. 1978).

123. Exceptional creativity or originality is not required; 'a modicum of creativity may suffice.' Universal Athletic Sales Co. v. Salkeld, 511 F.2d 904, 908 (3d Cir. 1974), cert. denied, 423 U.S. 863 (1975).

124. Both section 101 and the legislative history of the Act, see H.R. REP., supra note 56, at 53, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5666, make it clear that the seven categories in section 102 are only illustrative. The seven categories are: (1) literary works; (2) musical works; (3) dramatic works; (4) pantomimes and choreographic works; (5) pictorial, graphic, and sculptural works; (6) motion pictures and other audiovisual works; and (7) sound recordings. 17 U.S.C. section 102(a).

125. See H.R. REP., supra note 56, at 54, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5667. See generally Copyright Office Regulations, 37 C.F.R. section 201.20(g) (1983) (regarding copyright notices on machine-readable copies); id. section 202.20(c)(2)(VII) (regarding deposit of machine-readable work for registration purposes). These regulations imply that computer programs are covered by the 1976 Copyright Act.

126. See H.R. REP., supra note 56, at 51, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5664.

127. Pub. L. No. 93-573, 88 Stat. 1873 (1974) (codified at 17 U.S.C. section 701 note (1982)).

128. Final Report, supra note 3. See also 1 M. NIMMER, NIMMER ON COPYRIGHTS section 2.04(c), at 2-42 to 2-43 & n.21 (1983) (discussion of Final Report of CONTU).

129. See Final Report, supra note 3, at 55.

130. See M. NIMMER, supra note 128 section 2.04C, at 2-42 to 2-43 n.21. But see Final Report, supra note 3, at 86 (dissenting opinion of Commissioner Hersey arguing against copyrightability of computer programs).

131. Pub. L. No. 96-517, sections 10(a), 10(b), 94 Stat. 3015, 3028 (1980) (codified as amended at 17 U.S.C. sections 101, 117 (1982)).

132. See supra note 8 for the statutory definition of 'computer program.'

133. Section 117 essentially allows duplication or adaptation of a computer program during the actual utilization of the program. 17 U.S.C. section 117 (1982). This is an essential exception because when a program stored on a floppy disc, for example, is used by a computer, the machine must first copy the program from the disc into the computer's memory. Absent section 117, such copying would infringe a copyright on the program stored on a disc.

134. 17 U.S.C. sections 101, 102 (1982).

135. See H.R. REP., supra note 56, at 53, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5666.

136. White-Smith Publishing Co. v. Apollo Co., 209 U.S. 1 (1908), supports the argument that object code in a ROM is not copyrightable subject matter because the code is not directly perceivable to anyone. Nevertheless, the statutory language in section 102, which allows the work of authorship to be communicated 'with the aid of a machine or device,' effectively overruled the White-Smith decision. See H.R. REP., supra note 56, at 52, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5665 (legislative history clearly supports view that section 102 abrogates White-Smith).

137. Data Cash Syss., Inc. v. JS&A Group, Inc., 480 F.Supp. 1063, 1066-67 n.4 (N.D. Ill. 1979), aff'd, 628 F.2d 1038 (7th Cir. 1980).

138. Id.

139. Section 106 grants to the copyright owner the exclusive rights of reproduction, adaptation, publication, performance and display. 17 U.S.C. section 106 (1982).

140. See H.R. REP., supra note 56, at 61, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5675.

141. 2 M. NIMMER, supra note 128, section 8.01B, at 8-13. For instance, when an author publishes a book in hardcover the copyright law cannot be avoided by making a literal translation of the novel into a foreign language and then putting it on microfilm. Likewise, the analogous translation of source code into object code which is then permanently stored in a ROM does not avoid copyright protection granted for the source code.

142. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1251 (3d Cir. 1983).

143. See Mazer v. Stein, 347 U.S. 201, 217 (1954); Eden Toys, Inc. v. Marshall Field & Co., 675 F.2d 498, 500 (2d Cir. 1982); Miller v. Universal City Studios, Inc., 650 F.2d 1365, 1368 & n.1 (5th Cir. 1981); Gero v. Seven-Up Co., 535 F.Supp. 212, 216 (E.D.N.Y.), aff'd, 714 F.2d 113 (2d Cir. 1982); Atari, Inc. v. Amusements World, Inc., 547 F.Supp. 222, 228 (D. Md. 1981); Midway Mfg. Co. v. Dirkschneider, 543 F.Supp. 466, 479 (D. Neb. 1981); Dr. Pepper Co. v. Sambo's Restaurants, Inc., 517 F.Supp. 1202, 1207 (N.D. Tex. 1981).

144. See H.R. REP., supra note 56, at 57, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5670.

145. Compare Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1249 (3d Cir. 1983) (computer program in object code embedded in a ROM is copyrightable) with Data Cash Syss., Inc. v. JS&A Group, Inc., 480 F.Supp. 1063, 1066 n.4 (N.D. Ill. 1979) (computer program in object form in a ROM is mechanical device and therefore not copyrightable under the 1976 Copyright Act), aff'd, 628 F.2d 1038 (7th Cir. 1980).

146. See Midway Mfg. Co. v. Arctic Int'l Co., 547 F.Supp. 999, 1009 (N.D. Ill. 1982), aff'd, 704 F.2d 1009 (7th Cir. 1983). In affirming the lower court's decision in Midway, the Court of Appeals for the Seventh Circuit stated: 'Recording images and sounds in circuit boards does not destroy their copyrightability any more than does recording them on rolls of celluloid film.' 704 F.2d at 1012. Furthermore, even though a ROM contains a memory with a particular physical structure that may not be copyrightable without infringing the computer program embedded in the ROM, this limited monopoly is no greater than the monopoly created in the particular shape of the grooves in a record which represents an embodiment of a copyrighted work.

147. The test announced in Apple Computer requires a showing that the underlying idea cannot be expressed in more than one way before the idea and expression of the idea will be considered as merged. 714 F.2d at 1253.

148. See M. NIMMER, supra note 128, section 2.18C2, at 2-202.

149. Stern Elecs., Inc. v. Kaufman, 669 F.2d 825, 855 (2d Cir. 1982).

150. Apple Computer, 714 F.2d at 1253.

151. A microprocessor is an integrated circuit that includes a ROM and is the heart of a computer. It can be programmed with stored instructions to perform a wide variety of functions.

152. The Microcomputer Software Market: An Overview, COMPUTER SERVS. MONTHLY, NOV. 1982, at 1-8.

153. 714 F.2d at 1253. The Apple Computer test--under which a defendant must show that only one way exists to write an operating program for a particular computer--gives a defendant an opportunity to show that copyright protection is not available due to a merger of idea and expression. See supra text accompanying notes 119 & 120 for the Apple Computer court's application of this test.

154. See Final Report, supra note 3, at 75-78; Gates, Software and the Apple-Franklin Decision, Insurance for the Industry's Future, N.Y. TIMES, Sept. 25, 1983, section 3 (Business), at 2, col. 2.

155. The idea of granting a limited monopoly in return for the disclosure of scientific and literary achievements is the basis of U.S. CONST. art. I, section 8, cl. 8, which provides the authority for Congress to enact both the patent and copyright laws.

156. Note, Copyright Protection of Computer Program Object Code, 96 HARV.L.REV. 1723, 1739 (1983). Copyright Protection balances the author's rights against the interests of the general public. See White-Smith Music Publishing Co. v. Apollo Co., 209 U.S. 1, 19 (1908) (Holmes, J., concurring).

157. Midway Mfg. Co. v. Strohon, 564 F.Supp. 741, 751 (N.D. Ill.), cert. denied, 104 S.Ct. 90 (1983). Note, Copyright Protection of Computer Program Object Code, 96 HARV.L.REV. 1723, 1740 (1983). See also Note, Copyright Protection for Computer Programs in Read Only Memory Chips, 11 HOFSTRA L.REV. 329 (1982), in which the commentator states, 'without adequate protection, the economic incentive to produce new programs could disappear' and therefore 'to provide adequate protection for proprietary software and firmware, the law must be able to protect more than just direct copying.' Id. at 332, 361.

158. Software becomes fixed in a tangible medium of expression when, for example, the software is written down, printed out, stored on a disc or on magnetic tape, or embedded in a ROM.

159. 17 U.S.C. section 301(a) (1982); H.R. REP., supra note 56, at 129, reprinted in 1976 U.S. CODE CONG. & AD. NEWS 5659, 5745.

160. See 17 U.S.C. section 401 (1982) and Copyright Office Regulations, 37 C.F.R. section 201.20 (1983), which deal with the requirements for copyright notices.

161. 17 U.S.C. section 407 (1982). Registration, however, must be made prior to bringing a copyright infringement suit. Id. section 411.

162. Schmidt, Legal Proprietary Interests in Computer Programs: The American Experience, 21 JURIMETRICS J. 345, 375 (1981). Under the patent system, rights are only obtained after several years of proceedings before Office. See supra note 37 and accompanying text for a 4, col. 1 (Copyright Office data shows discussion of the time required to obtain a patent.

163. See Brieff, Computer Boom Leads to Copyright Emphasis, LEGAL TIMES, Dec. 6, 1982, at 4, col. 1 (Copyright Office data shows that 40% of registered programs are written by individuals or very small businesses). See supra note 36 for discussion of the cost of obtaining a patent.

164. Trade secret law provides monopoly for as long as the proprietary information is not publicly disclosed. The patent law provides a monopoly for 17 years for utility patents. 35 U.S.C. section 154 (1982).

165. The importance of the need to protect software is exemplified by a bill introduced in the House of Representatives on May 19, 1983, which would 'strengthen the laws against computer program and data base piracy and counterfeiting.' H.R. 6420, 97th Cong., 2d Sess. (1982).

166. See Gates, supra note 154 (all software must be protected to provide an incentive for development of new software). See also Borden, Endorsing A Computer Monopoly, N.Y. TIMES, Sept. 25, 1983, section 3 (Business), at 2, col. 2, in which the author, chief executive of Franklin Computer Corp., writes that copyright protection of software is proper, but that such protection should not extend to an operating system program that is an integral part of the computer.

167. For discussion of the cases affording copyright protection to computer software, see supra note 64 and accompanying text.

168. See supra notes 71 & 80-93 and accompanying text for a survey of cases applying copyright law to software.

169. See supra note 71 for a survey of cases limiting the scope of copyright protection for software.

170. 174 F.2d at 1250.

171. Id. at 1253.