Copyrightability of Software: Piracy on the Waters of Protection

Sylvia Ann Matthews

Follow this and additional works at: https://scholarcommons.sc.edu/sclr

Part of the Law Commons

Recommended Citation
Available at: https://scholarcommons.sc.edu/sclr/vol37/iss4/6

This Note is brought to you by the Law Reviews and Journals at Scholar Commons. It has been accepted for inclusion in South Carolina Law Review by an authorized editor of Scholar Commons. For more information, please contact dillarda@mailbox.sc.edu.
NOTES

COPYRIGHTABILITY OF SOFTWARE:
PIRACY ON THE WATERS OF PROTECTION

I. INTRODUCTION

Technological developments and persuasive marketing techniques have made computers more popular in recent years, and as the use of computers has increased, so has the incidence of software piracy. This Note describes the dilemma facing producers of original software, examines the development of copyright protection for computer software, and explores the parameters of that protection by considering common problems in copyright law applicable to computer software.

“Piracy” of software occurs most frequently in two situations. First, infringement occurs when software that is produced by one company is copied and sold by another company or incorporated into the infringing company’s products. Second, misuse occurs when lawful users of the program make unauthorized copies of a particular program.


2. The term “software” may be used to refer to computer programs, data bases, or documentation. Bender, *Software Protection: The 1985 Perspective*, 7 W. New Eng. L. Rev. 405, 407 (1985). A “program” is a “set of precise instructions that tells the computer how to solve a problem.” Data Cash Sys., Inc. v. JS&A Group, Inc., 480 F. Supp. 1063, 1065 (N.D. Ill. 1979) (citations omitted). For the statutory definition of “computer program” and its history, see infra note 57 and accompanying text. A data base is a grouping of information on which the machine performs its operations. Documentation refers to the comments that are included in a program to explain its operation. Bender, supra, at 407. This Note will use “software” to mean computer programs.
A. Piracy by Competitors

The software age began in 1970 when International Business Machines (IBM) began selling software and hardware separately. Since its inception, investments in the software industry have soared. According to one commentator, "More money is spent on the development of software today than on any other single technological asset." The same writer estimated the value of domestic software in 1978 at over seventy billion dollars. One of the leading microcomputer manufacturers, Apple Computer, Inc., may have spent as much as fifty million dollars in research and development of original software.

The most crippling effect of competitor piracy is financial; companies that invest large amounts of capital to develop original software do not reap the projected benefits when competing companies copy the programs and market them at a price below that charged by the developing company. Franklin Computer Corp., for example, copied programs developed by Apple and was then able to sell them at four hundred dollars below the cost of Apple's system. Mail order companies have also hurt original producers' business by selling pirated copies of software for as little as half the cost of the original. This severe competition in the software market may have contributed to the depression experienced in the computer industry during 1985.

5. Milgrim, supra note 4, at 158.
6. Id.
7. Getting Tough on Software Theft, supra note 1, at 28.
10. During the summer of 1985, Wang, Burroughs, and Hewlett-Packard announced the layoffs of thousands of employees. Lueck, Boston Looks Past Computers, N.Y. Times, July 29, 1985, at D1, col. 3; Sanger, Computer Makers are Waiting to Hit Bottom, N.Y. Times, July 21, 1985, at E7, col. 4; N.Y. Times, July 19, 1985, at D4, col. 6. IBM suffered a 15% reduction in profits during the first half of 1985. After a summer slump, the computer industry initiated a yearend Christmas campaign to boost fourth quarter sales and salvage the year. Lewis, Guess What Computer Makers Want for
Estimates of the marketshare that is lost to bootleggers range from thirty to fifty percent. According to a Dallas-based research firm, one pirated copy exists for every authorized copy of business software, and piracy cost the software industry 1.3 billion dollars in lost revenues between 1981 and 1984.

Software companies, in efforts to preserve their investments, have resorted to technological devices such as the "serial number plan." A manufacturer using this technique would install a unique serial number on each program, which could then be used only in a machine with a corresponding serial number. This method would be effective in curbing user piracy as well as competitor piracy: "Users would make unlimited back-up copies but could not pass them on because neither the machine nor the software would operate unless the numbers matched." This plan, however, presents at least two problems. First, new software would be needed everytime the hardware was upgraded or changed. Second, the bundling of hardware products with software products might constitute a tying arrangement in violation of the antitrust laws because of its effect on competition among other software producers. Another proposed method of deterrence would program popular software into computer chips that would be installed in the hardware when sold. Like the serial number plan, however, this proposal might thwart competition and make upgrading hardware difficult.

For lack of a better remedy, computer companies have turned to litigation to prevent copying of their software. In the

---

References:

15. User piracy is discussed infra notes 21-30 and accompanying text.
17. Id.
past few years, numerous complaints alleging copyright violations have been filed against competitors. Only a few, however, have reached the trial stage. Three of these cases will be discussed below, but more litigation can be expected unless competitor piracy ceases.

B. Piracy by Users

A user pirate is one who is in legal possession of a program and copies it, usually for his personal convenience. User piracy often begins innocently, as the following excerpt from a data processing trade magazine illustrates:

Your company has acquired a spreadsheet computer program for the vice president of sales to use on his desktop pc [personal computer]. He keeps the program diskette on the bookshelf in his office. An employee from across the hall borrows the diskette the same way he would borrow a book. That employee then makes a copy of the diskette to use with his own computer so he won't have to keep borrowing the vice president's copy. He also makes a copy for the person at the next desk to use on his pc, and another one that he mails to his counterpart in one of the branch offices. Finally, the employee makes an extra copy to take home so he can complete his calculations on his own pc or let family members use it to do their work.

Despite the seeming innocence of this misuse, producers of original programs are taking a tough stand against their customers. For example, Lotus Development Corp., the nation's largest manufacturer of software for personal computers, filed suit


22. Id.

23. Lueck, supra note 10, at D6, col. 5.
against its customer, Rixon Corporation, to prevent the unauthorized copying of Lotus computer programs for distribution to Rixon’s branch offices.\textsuperscript{24} The suit was settled out of court “for an undisclosed, but reportedly substantial amount.”\textsuperscript{25}

Lotus also sued The Health Group, Inc., alleging user piracy by both the corporate entity and the manager of the corporation’s data processing department.\textsuperscript{26} Lotus’ apparent strategy is to impose personal responsibility for infringement violations on individuals within a corporation.\textsuperscript{27}

The Association of Data Processing Service Organizations, Inc. (Adapso) has also developed a policy against user pirates. In January 1985 Adapso teamed up with Micropro to sue American Brands, Inc. and its subsidiary, the Wilson Jones Co., for unauthorized copying.\textsuperscript{28} This attack on software users is aimed at enforcing the producer’s copyrights and informing the customers that the producers are serious about preventing unauthorized copying.\textsuperscript{29}

Software producers are also developing techniques to deter unauthorized customer copying. Adapso, for example, is considering the use of “keys” issued with purchased packages. Mounted on one microcomputer would be “keyrings,” which could accommodate as many as sixteen “keyways,” allowing the operator to use many copy-protected packages simultaneously. This approach has not yet been adopted, however, because of fear that keyrings would be a burden to users. A less intrusive alternative would be for the producer to collect a small fee for each additional program user,\textsuperscript{30} but since enforcement of this alternative would be difficult, if not impossible, its value is questionable.

The problem that software manufacturers face is twofold: competitors pirate original products to sell as their own, and

\textsuperscript{24} Smiddy & Smiddy, supra note 12, at 102. Lotus sued Rixon Corp. of Maryland for $10,000,000 for making and distributing unauthorized copies to their branch offices. Lotus also sued The Health Group, Inc. of Tennessee on the same grounds and reached a settlement for an undisclosed amount.

\textsuperscript{25} Id.

\textsuperscript{26} Braue, supra note 11, at 16.

\textsuperscript{27} Id. at 18.

\textsuperscript{28} Smiddy & Smiddy, supra note 12, at 102.

\textsuperscript{29} For statutorily authorized copying, see infra notes 59-61 and accompanying text.

\textsuperscript{30} Vendors Step Up Anti-Copying Campaign, COMPUTER DEC., Feb. 26, 1985, at 1986.\textsuperscript{683}
customers make and distribute unauthorized copies. Technological devices may deter some software piracy, but judicial intervention is necessary for lasting enforceable protection. Parts II through V of this Note discuss the development of copyright protection for software.

II. STATUTORY PROVISIONS

Pursuant to article 1, section 8, clause 8 of the Constitution,31 Congress replaced the 1909 Copyright Act by enacting the Copyright Act of 1976 (the 1976 Act).32 According to the 1976 Act, a work may be protected if it is "an original work of authorship fixed in any tangible medium of expression."33 The legislative history of section 102 states that the "two fundamental criteria of copyright protection . . . [are] originality and fixation in tangible form."34 According to section 102, works of authorship include the following: "(1) literary works; (2) musical works, including any accompanying words; (3) dramatic works, including any accompanying music; (4) pantomimes and choreographic works; (5) pictorial, graphic, and sculptural works; (6) motion pictures and other audiovisual works; and (7) sound recordings."35 The legislative history of the 1976 Act suggests that the category of "literary works," as defined in section 101,36 includes "computer data bases, and computer programs to the extent that they incorporate authorship in the programmer's expression of original ideas, as distinguished from the ideas themselves."37

The requirement of fixation is met if the work "can be per-

31. U.S. CONST. art. I, § 8, cl. 8 empowers Congress to lay and collect taxes as necessary "[t]o promote the progress of sciences and useful arts, by securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries."


35. 17 U.S.C. § 102(a) (1982). This list, however, is not intended to be exclusive. For an explanation of "including," see 17 U.S.C. § 101 (1982).

36. 17 U.S.C. § 101 (1982) defines "literary works" as "works, other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects, such as books, periodicals, manuscripts, phonorecords, film, tapes, disks, or cards, in which they are embodied."

ceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device." The medium involved need not be of any particular form. The expression may be "in words, numbers, notes, sounds, pictures and any other graphic or symbolic indicia, whether embodied in a physical object in written, printed, photographic, sculptural, punched, magnetic, or any other stable form." Section 102 specifically states that a protected expression may be "perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device." The legislative history of the 1976 Act, however, also indicates that fixation may not be achieved if the expression is merely "captured momentarily in the 'memory' of a computer."

Subsection (b) of section 102 provides that copyright protection extends only to the expression of an original work and does not extend to the underlying ideas on which the expression may be based. The legislative history reveals concern over the extension of copyright protection of a computer program to the "methodology or processes adopted by the programmer, rather than merely to the 'writing' expressing his ideas." The legislative history also repeats that the purpose of section 102 is to limit protection to the "expression" adopted by the programmer. Although the 1976 Act does not specifically provide copyright protection for computer programs, it may be implied from the legislative intent.

38. 17 U.S.C. § 102(a) (1982). By using broad language, Congress intended to avoid problems created by cases such as White-Smith Publishing Co. v. Apollo Co., 209 U.S. 1 (1908). The Court in White-Smith held that a piano roll was not a copy of the musical composition because it was not in a form that others could easily "see and read." See 1 M. Nimmer, Nimmer on Copyright § 2.03[B][1] (1983).
42. "In no case does copyright protection for an original work of authorship extend to any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." 17 U.S.C. § 102(b) (1982).
44. Id.
45. Although it may appear that Congress intended to protect computer programs,
In 1974, in preparation for passage of the 1976 Act, Congress created a National Commission on New Technological Uses of Copyrighted Works (CONTU)\textsuperscript{46} to study, \textit{inter alia}, the use of copyrighted works in computers.\textsuperscript{47} After three years of study, CONTU submitted its final report, in which it suggested three changes affecting software copyright. First, CONTU suggested amending the 1976 Act "to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright."\textsuperscript{48} CONTU also proposed that the law be amended "to apply to all computer uses of copyrighted programs . . . and . . . to ensure that rightful possessors of copies of computer programs can use or adapt these copies for their use."\textsuperscript{49} To achieve these objectives, CONTU recommended amending section 101 of the copyright law to include a definition of "computer program" and drafting a new section 117.\textsuperscript{50}

Because Congress had initially recognized the need to address computer copyright problems in 1976, but was uncertain of the appropriate method, the original version of section 117 provided:

\textsuperscript{46} CONTU's first priority was to make the intent explicit.


\textsuperscript{48} Specifically, the Commission was empowered "to study and compile data on the reproduction and use of copyrighted works of authorship . . . in conjunction with automatic systems capable of storing, processing, retrieving, and transferring information . . . and to make recommendations as to such changes in copyright law or procedures that may be necessary." Pub. L. No. 93-573, \textsection 201, 88 Stat. 1873 (1974).

\textsuperscript{49} \textit{CONTU Final Report}, \textit{supra} note 46, at 1.

\textsuperscript{50} Id. at 12.
Notwithstanding the provisions of sections 106 through 116 and 118, this title does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information, or in conjunction with any similar device, machine, or process, than those afforded to works under the law, whether title 17 or the common law or statutes of a State, in effect on December 31, 1977, as held applicable and construed by a court in an action brought under this title.\textsuperscript{51}

Thus, this section "provided that the law on the use of copyrighted works in computer systems would be unaffected by its enactment."\textsuperscript{52} The purpose of section 117 was to preserve the "status quo" until CONTU concluded its study and filed its final report.\textsuperscript{53}

CONTU concluded that section 117, as written, made "clear that the placement of any copyright work into a computer is the preparation of a copy and, therefore, a potential infringement of copyright."\textsuperscript{54} To provide that persons in lawful possession of copies of programs could use them without fear of infringement, CONTU drafted a new version of section 117, which provided:

Notwithstanding the provisions of section 106, it is not an infringement for the rightful possessor of a copy of a computer program to make or authorize the making of another copy or adaptation of that computer program provided:

(1) that such a new copy or adaptation is created as an essential step in the utilization of the computer program in conjunction with a machine and that it is used in no other manner, or (2) that such new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were pre-

\begin{itemize}
  \item 52. CONTU Final Report, supra note 46, at 4.
  \item 54. CONTU Final Report, supra note 46, at 12.
\end{itemize}
pared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.\(^5^5\)

Congress adopted CONTU's proposed changes almost verbatim.\(^5^6\) In 1980 Congress amended section 101 to include a definition of "computer program"\(^5^7\) and repealed old section 117 replacing it with the version proposed in the CONTU report.\(^5^8\)

Although a copyrighted program generally may not be duplicated without the developer's permission, the current version of section 117 provides two exceptions to this rule. First, the user may load a program into the internal memory of the computer to use it. "At the time of use, there are two copies of the program: the one in the diskette and the one executing in memory."\(^5^9\) Section 117 specifically provides that the copy in memory is not an infringing copy. The second exception provides that one in rightful possession may prepare archival copies to guard against destruction, but may not sell some copies, while retaining others.\(^6^0\) If the rightful possessor sells copies of a program, the sale "must be of all rights in the program, thus creating a new rightful possessor and destroying that status as regards the seller."\(^6^1\)

Although Congress adopted all of the CONTU proposals, CONTU was not in unanimous agreement about the degree of protection that should be accorded computer programs.\(^6^2\) In a concurring opinion, Commissioner Nimmer stated that the majority's proposals were "open-ended" to the point that "if 'liter-

---

55. Id. For a discussion of § 117 and its history, see Stern, Section 117 of the Copyright Act: Charter of the Software Users' Rights or an Illusing Promise, 7 W. New Eng. L. Rev. 459 (1985).
56. In the version of § 117 that Congress adopted, "rightful possessor" was changed to "owners." See 17 U.S.C. § 117 (1982).
57. The new definition states: "[A] 'computer program' is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about certain results." Id. § 101 (1982).
59. Smiddy & Smiddy, supra note 12, at 104.
62. CONTU Final Report, supra note 46, at 12.
ary works' are to be so broadly construed, the Copyright Act becomes a general misappropriation law." According to Nimmer, the constitutionality of the majority's proposals could be challenged because such an approach arguably stretches the meaning of 'authors' and 'writing' beyond constitutional limits. He concluded that the CONTU proposals might be "unduly restrictive" and proposed his own "demarcation which would distinguish between protectible and nonprotectible software."

Nimmer proposed that copyright protection for software be limited "to those computer programs which produce works which themselves qualify for copyright protection." As an illustration of this proposal he suggested that "[a] program designed for use with a database . . . would clearly be copyrightable since the resulting selection and arrangement of items from such database would be copyrightable as a compilation." For instance, programs designed for use in conjunction with legal information or a computer game would be copyrightable since the result or output would constitute, respectively, an enumeration of cases or an audiovisual work, each entitled to copyright protection. Nimmer suggested that these protected programs differ from a program that controls the flow of traffic, which would "not be eligible for copyright because its operations do not result in copyrightable works." Analogyizing this medium to sound recordings, Nimmer concluded: "[T]he operation of the sound recording produces a musical work which itself is copyrightable. That is sufficient to render the sound recording itself copyrightable quite apart from the separate copyright in the musical work." Nimmer thus proposed to disregard the form of the program and look exclusively to the result of the program's operation. He acknowledged, however, that his suggestion, although useful for future implementation, might not be appropriate at present.

63. Id. at 26.
64. Id. For a discussion of the historical uses of copyright, see the dissent of Commissioner Hersey, id. at 27.
65. Id. at 27.
66. Id.
67. Id.
68. Id.
69. Id.
70. Id.
The majority considered and specifically rejected Nimmer's proposal:

It has been suggested by Vice-President Nimmer in his separate opinion that programs be copyrighted only when their use leads to copyrighted output. . . . This distinction is not consistent with the design of the Act of 1976, which was clearly to protect all works of authorship from the moment of their fixation in any tangible medium of expression. Further, it does not square with copyright practice past and present which recognizes copyright protection for a work of authorship regardless of the uses to which it may be put. The copyright status of the written rules for a game or a system for the operation of a machine is unaffected by the fact that those rules direct the actions of those who play the game or carry out the process. Nor has copyright been denied to works simply because of their utilitarian aspects. It follows, therefore, that there should likewise be no distinction made between programs which are used in the production of further copyrighted works and those which are not.71

Commissioner Hersey, in a dissenting opinion,72 recommended that “[t]he Act of 1976 should be amended to explicitly state that copyright protection does not extend to a computer program capable of being used to control computer operations.”73 According to Hersey, a “[computer] program itself, in its mature and usable form, is a machine control element, a mechanical device, which on Constitutional grounds and for reasons of social policy ought not to be copyrighted.”74 Hersey maintained that since protection for computer programs existed or would exist in other areas of law, copyright protection was not needed and, if given, would have negative social and economic consequences.75

Hersey conceded that the products of preparatory stages of

71. Id. at 21 (footnote omitted).
72. Commissioner Hersey’s dissent persuaded Commissioner Karpatkin to dissent as well. Id. at 38.
73. Id. at 1. He suggested three possible boundaries for copyright protection: “(1) the moment of transformation from ‘source’ to ‘object’ program; (2) the moment of input into a computer or micro-processor; or (3) at the point where a program goes from ‘natural language’ . . . to higher-level, formal computer language.” Id. at 37.
74. Id. at 27.
75. Id. at 27-28.
programming, such as flowcharting and source code,\textsuperscript{76} are "writings," but argued that a computer program in its mature phase is a mechanical device that, when activated, performs a task.\textsuperscript{77} To stress the instructional nature of programs, he analogized a computer program to a motorized cam: "A cam, like a mature computer program is the objectification of a series of instructions: 'Up, down, up, down . . .' or 'In, out, in, out . . . .'\textsuperscript{78} Both embody instructions that result in the production of work and both were originally "conceptualized, described, and written out as . . . [a] series of instructions."\textsuperscript{79} Hersey also implied that it would be ludicrous for anyone to suggest, as they had for computer programs, that a cam is "a literary work" deserving copyright protection.\textsuperscript{80}

Hersey's dissent voiced two fundamental disagreements with the majority. First, he argued that because copyright protection has never been extended to a form of expression that is not perceivable by the human senses,\textsuperscript{81} it should not be extended to computer programs. Second, he expressed fear that the majority's proposals extended protection beyond the expression to the underlying idea.\textsuperscript{82}

III. CASE LAW CONCERNING COPYRIGHT PROTECTION FOR SOFTWARE

The Commission's proposals and the subsequent 1980 amendments to the 1976 Act have provided substantial guidance for courts considering claims of software infringement. Two circuit courts of appeal have interpreted the recent changes to mean that virtually all computer programs are entitled to protection.\textsuperscript{83} A review of those two cases, and others that preceded them, will reveal the development and current level of copyright

\textsuperscript{76} For an explanation of "source code," see infra note 86.
\textsuperscript{77} Id. at 28.
\textsuperscript{78} Id. at 29.
\textsuperscript{79} Id.
\textsuperscript{80} Id. at 29-30.
\textsuperscript{81} Id. at 32-33, 36-37.
\textsuperscript{82} Id. at 33.
protection available for computer software.

One of the first cases considering copyright infringement of a computer program was Data Cash Systems, Inc. v. JS&A Group, Inc. In Data Cash the plaintiff alleged that the defendant infringed its copyright on a computer chess game by marketing an identical game. Apparently, the third party that developed the plaintiff's game duplicated for the defendants the plaintiff's ROM, which contained the program in its object code form. The plaintiff filed a motion for a preliminary injunction and the defendant sought summary judgment. The court ruled that section 117, as it existed in its original form, precluded the use of the 1976 Act to protect the alleged copyright. The court relied, therefore, solely on the 1909 Act and common-law copyright provisions.

Although the court found that the object programs in both products were identical and, therefore, assumed that copying had occurred, it found no infringement because the medium to

84. 480 F. Supp. 1063 (N.D. Ill. 1979), aff'd on other grounds, 628 F.2d 1038 (7th Cir. 1980).

85. The ROM (Read Only Memory) is an internal permanent memory device, consisting of a semi-conductor "chip," that is incorporated into the circuitry of the computer. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1243 (1983).

86. There was no copyright notice on the ROM, but the copyright notice did appear on the source program. 480 F. Supp. at 1066. In Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (1983), cert. dismissed, 464 U.S. 1033 (1984), the court provided the following explanation of the terms "object code" and "source code":

There are three levels of computer language in which computer programs may be written. High level language, such as commonly used BASIC or FORTRAN, uses English words and symbols, and is relatively easy to learn and understand (e.g. "Go to 40" tells the computer to skip intervening steps and go to the step at line 40). A somewhat lower level language is assembly language, which consists of alphanumeric labels (e.g. "ADC" means "add with carry"). Statements in high level language and apparently also statements in assembly language, are referred to as written in "source code." The third, or lowest level computer language, is machine language, a binary language using two symbols, 0 and 1, to indicate an open or closed switch (e.g. "01101001" means, to the Apple, add two numbers and save the result). Statements in machine language are referred to as written in "object code."

Id. at 1243. For a brief, but clear explanation of computer programs terminology, see Note, Copyright Protection of Computer Program Object Code, 96 Harv. L. Rev. 1723, 1724-26 (1983).

87. 480 F. Supp. at 1065.

88. Id. at 1067.

89. The defendant stipulated that its chess computer program was identical to the plaintiff's. Id. at 1066 n. 3.

90. Id. at 1066.
which the program was copied did not constitute a "copy." According to the court, "Both at common law and under the 1909 Act, a 'copy' must be in a form which others can see and read."\textsuperscript{91} The court found that the ROM could not be seen or read by others and, thus, was not a copy.\textsuperscript{92} The court also found that "[i]n its object phase, the ROM, the computer program is a mechanical tool or a machine part but it is not a 'copy' of the source program."\textsuperscript{93} As a result, the court concluded that the defendant's program, which was embodied in the ROM, did not infringe upon the plaintiff's copyright and granted the defendant's motion for summary judgment.\textsuperscript{94}

In *Tandy Corp. v. Personal Micro Computers, Inc.*,\textsuperscript{95} the defendant also made a motion to dismiss the plaintiff's copyright infringement claims, but the court denied the motion. The defendant argued that when a computer program is stored on a silicon chip, a copy of the chip does not infringe the program that is embodied on the chip. According to Tandy, the defendant had copied the Tandy TRS-80 compiler program and then

\textsuperscript{91} Id. at 1068.
\textsuperscript{92} Id. at 1069. To reach its conclusion under the 1909 Act, the court relied on the Supreme Court's decision in White-Smith Music Publishing Co. v. Appollo Co., 209 U.S. 1 (1908)(piano roll for player piano found not to be a copy of the musical composition contained on it). The object program, like the piano roll, was not in a form that could be seen or read with the naked eye and, thus, was found not to be a copy. 480 F. Supp. at 1069. According to the court, the common-law definition of copy is "that which comes so near to the original as to give to every person seeing it the idea created by the original." Id. at 1068 (footnotes omitted). The court analogized a ROM to a building that is built according to architectural plans. The court concluded that an architectural plan is a writing that can be copied, but a building built pursuant to the plans is not a "copy" of the plans. *Id.*
\textsuperscript{93} 480 F. Supp. at 1069 (footnote omitted). See *supra* note 92.
\textsuperscript{94} The *Data Cash* court has been criticized for its misuse of terminology and confusion of object code with the ROM in which it is embodied. See generally D. BENDER, COMPUTER LAW, § 4.04[1], at 4-22 to -23 (1983). The court also misapplied the definition of "copies" found in 17 U.S.C. § 102(a) (1982). See *supra* note 38 and accompanying text. According to the court, "[E]ven if the 1976 Act did apply, copying of the ROM would not be actionable." 480 F. Supp. at 1066 n. 4. The legislative history of § 102, however, expressly provides that the seeing and reading requirement enunciated in *White-Smith* is to be rejected. See *supra* note 38. On appeal the Seventh Circuit affirmed on other grounds. Since the lower court's rationale was not argued or briefed by either side, the appellate court did not consider it. 628 F.2d at 1041. See generally 2 M. Nimmer, *supra* note 38, § 8.08 n. 18 (stating that the Seventh Circuit implicitly rejected the district court's reasoning).
\textsuperscript{95} 524 F. Supp. 171 (N.D. Cal. 1981).
used it in its own computer. The court found that the program was of a copyrightable nature and in a fixed form on the chip. Disagreeing with the Data Cash court, the Tandy court concluded that a chip is a "copy" within the meaning of sections 101 and 102 of the 1976 Act and that, therefore, the program was within the copyright provisions of section 102.

In GCA Corp. v. Chance, the United States District Court for the Northern District of California considered whether the "object code" version of a program is protected when the "source code" version has been registered and properly copyrighted. The court granted GCA's motion for a preliminary injunction based on GCA's prima facie showing of a valid copyright of the source code, Chance's access to the copyrighted works, and Chance's admission that it had copied the operating programs. According to the court, because the object code is merely a different version of the source code, "the two are to be treated as one work." Thus, when the source code is protected, so is the object code.

In Apple Computer, Inc. v. Franklin Computer Corp., Apple sought to enforce its copyrights of operating systems programs that were expressed in object code. The Court of Ap-

---

96. Id. at 173.
97. Id. Additionally, the court denied the defendant's motion to dismiss because the plaintiff suggested that the chip might have been duplicated by first copying the printout or the visual display. Id. at 175. According to the court, if proven, "there can be no doubt that the unauthorized duplication of a visually displayed copy of the program would fall within the reach of the federal copyright laws." Id.
98. 624 F. Supp. at 174-75.
100. For an explanation of "object code" and "source code," see supra note 86.
101. 1982 Copyright L. Dec. at 17,764-65. The court stated that to prevail on the copyright infringement claim, the "plaintiff must establish (1) ownership of a valid copyright; (2) access to the copyrighted works; and (3) copying by showing substantial similarity as to the general idea contained in the works." Id. (citing Sid & Marty Krofft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157, 1164 (9th Cir. 1977)).
102. 1982 Copyright L. Dec. at 17,765.
103. Id.
105. Most programs fall into one of two general categories. "Application programs" perform a specific task for the user, such as updating a file or printing a list. "Operating systems programs" generally manage the internal functions of the computer or facilitate use of application programs.
106. 714 F.2d at 1243. For an explanation of "object code," see supra note 86.
peals for the Third Circuit addressed two issues: whether computer programs expressed in object code are copyrightable and whether an operating systems program is entitled to copyright protection.  

Franklin manufactured the ACE 100 personal computer and, in order to make the ACE 100 more marketable, designed it to be "Apple compatible." Thus, the software designed for Apple computers could also be used in ACE 100. Franklin did not deny that it had copied Apple's programs, but claimed to have done so to ensure Apple compatibility. Apple alleged that Franklin was liable for patent infringement, unfair competition, misappropriation, and copyright infringement of fourteen of Apple's computer programs. Franklin asserted, as an affirmative defense, that the Apple programs were not copyrightable subject matter. The district court denied Apple's original motion for a preliminary injunction enjoining Franklin from infringing the copyrights Apple held for certain computer programs.

Three days after the date of the district court's order, the Third Circuit decided Williams Electronics, Inc. v. Artic International, Inc. In Williams Electronics the court considered the copyrightability of a computer program for a video game that was embodied in a ROM and concluded that "the copyrightability of computer programs is firmly established after the 1980 amendment to the Copyright Act." The court then held that Artic had failed to show "any persuasive reason which would overcome the statutory presumption of validity of the


108. 714 F.2d at 1243.

109. Id. at 1245. An Apple systems programmer testified that Franklin must have copied because "it is 'almost impossible for so many lines of code' to be identically written," and because the Franklin master disk contained his name and another word that he had embedded in certain Apple programs. Id.

110. Id. at 1243.

111. Id. at 1244.

112. 685 F.2d 870 (3d Cir. 1982). The Franklin order and opinion were dated July 30, 1982. 714 F.2d at 1245. Williams Electronics was decided on Aug. 2, 1982. 685 F.2d at 870.

113. For a definition of "ROM," see supra note 85.

114. 685 F.2d at 875.
[computer program's] copyright registration" and affirmed the grant of an injunction against Artic. In light of the Williams Electronics decision, Apple petitioned the district court to reconsider its claim. The court denied Apple's motion for reconsideration, and Apple appealed.

Replying to Franklin's first assertion, that object code is not proper copyright subject matter, the court of appeals stated that there was "no basis in the statute for any such concern." After an examination of the Copyright Act's legislative history and of the CONTU Report, the court concluded that "a computer program, whether in object code or source code, is a 'literary work' and is protected from unauthorized copying, whether from its object or source code version.

Franklin also claimed "that computer operating systems programs, as distinguished from application programs, are not the proper subject of copyright 'regardless of the language or medium in which they are fixed.' Franklin contended that systems programs are not copyrightable because they are either a "process," "system," or "method of operation" and section 102(b) of the Act states that copyright protection does not extend to "any idea, procedure, process, [or] system." After noting that Franklin had raised an issue of first impression, the court concluded that Apple was not claiming copyright protection to the underlying process, but only to the expression of the instructions. The court also noted that the distinction Franklin sought to make was difficult to reconcile with its concession that application programs are copyrightable subject matter. Since only the computer instructions were protected, the court could

115. Id. at 877.
116. 714 F.2d at 1245.
117. Id at 1247. The copyrightability of object code was also at issue in Hubco Data Prods. Corp. v. Management Assistance, Inc., 219 U.S.P.Q. (BNA) 450 (D. Idaho 1983). Hubco developed a procedure for altering the memory and function capabilities of systems installed by Management Assistance, Inc. (MAI). Id. at 452. In issuing a preliminary injunction against Hubco, the court relied on Williams Electronics and the CONTU final report to find that an object code is copyrightable. 219 U.S.P.Q. at 454.
118. 714 F.2d at 1249 (footnotes omitted). The court based its decision on the statutory definition of literary works, see supra note 36, and a provision in the CONTU Report majority opinion stating that "object codes are proper subjects of copyright." 714 F.2d at 1248.
119. Id. at 1249 (citing Brief of Appellee at 15).
120. Id. at 1250-51.
121. Id. at 1250.
not distinguish between "instructions in an operating system program . . . [and] the instructions in an application program." 122

The court also rejected Franklin's argument that the program becomes an actual part of the machine and, thus, should not be protected. The court cited the CONTU report and the Commission's conclusion that programs are not machine parts, just as video tapes are not part of the video machine, and concluded that the program does not become part of the machine. 123

Franklin also challenged the copyrightability of the programs because of the utilitarian nature of operating systems. 124 The court, however, cited the Supreme Court's statement, in Mazer v. Stein, 125 that the intended use of an article should not determine copyrightability and the CONTU majority's opinion that the use of a program to perform a process does not affect copyrightability 126 and rejected this argument as well. 127

Finally, Franklin argued that extending protection to systems programs would constitute protection of an idea. The defendant cited Mazer v. Stein 128 and section 102(b) of the 1976 Act for the proposition that copyright protection extends only to the expression of the idea, not the idea itself, and argued that systems programs are, therefore, not copyrightable. 129 The court responded by formulating the following test to distinguish the expression of a systems program from the underlying idea: "If other programs can be written or created which perform the same function as an Apple's operating system program, then that program is an expression of the idea and hence, copyrightable." 130 Therefore, to prove that a systems program is copyrightable, a party need only present competitors' products that are written differently, but perform the same functions. The court instructed the district court to make findings regarding the

122. Id. at 1251.
123. Id. (citing CONTU Final Report, supra note 46, at 21).
124. Id. The argument that "purely utilitarian works" are not proper copyright subject matter was derived from dicta in Baker v. Seldin, 101 U.S. 99, 103 (1879).
126. 714 F.2d at 1252 (citing CONTU Final Report, supra note 46, at 21).
127. 714 F.2d at 1252.
128. 347 U.S. at 217.
130. 714 F.2d at 1253 (citing Dymow v. Bulton, 11 F.2d 690, 691 (2d. Cir. 1926)).
existence of other programs if the issue was pressed on remand.\textsuperscript{131}

In Apple Computer, Inc. \textit{v. Formula International, Inc.},\textsuperscript{132} Formula appealed from the district court's grant of a preliminary injunction prohibiting Formula from copying, distributing, or selling Apple's copyrighted computer programs and from using the name "Pineapple" as its trademark or trade name. The computer programs at issue in \textit{Formula} were operating systems programs\textsuperscript{133} that were registered pursuant to the 1976 Copyright Act.\textsuperscript{134} Formula conceded that its programs were substantially similar to Apple's,\textsuperscript{135} but asserted that "because . . . [the systems programs] control the internal operation of the computer, [they] are only 'ideas' or 'processes,' and therefore, unlike application programs, . . . are not protected by copyright."\textsuperscript{136} Formula also claimed that under the idea/expression dichotomy, "a computer program is protected under the Copyright Act only if the program embodies expression which is communicated to the user when the program is run on a computer."\textsuperscript{137}

The court of appeals found Formula's claims contrary to the language of the Copyright Act, its legislative history, and the existing case law.\textsuperscript{138} After reviewing the CONTU report, including Nimmer's and Hersey's separate opinions, the court concluded that the 1976 Act "makes no distinction between the copyrightability of those programs which directly interact with the computer user and those which simply manage the computer system."\textsuperscript{139} The court ruled that the idea/expression argument failed because the idea involved could be expressed in numerous ways and Apple merely "[sought] to copyright . . . its particular set of instructions, not the underlying computer process."\textsuperscript{140}

The trend for the past few years has clearly been to broaden

\begin{itemize}
\item \textsuperscript{131} 714 F.2d at 1253.
\item \textsuperscript{132} 562 F. Supp. 775 (C.D. Cal. 1983), aff'd, 725 F.2d 521 (9th Cir. 1984). For a thorough treatment of this case, see Case Comment, Copyright Law-Computer Programs, 23 Duq. L. Rev. 467 (1985).
\item \textsuperscript{133} For an explanation of "operating systems programs," see supra note 105.
\item \textsuperscript{134} 725 F.2d at 523.
\item \textsuperscript{135} Id.
\item \textsuperscript{136} Id.
\item \textsuperscript{137} Id. at 524.
\item \textsuperscript{138} Id.
\item \textsuperscript{139} Id. at 525.
\item \textsuperscript{140} Id.
\end{itemize}
the scope of protection for computer programs. After the *Franklin* and *Formula* decisions, it appears that application and systems programs are copyrightable in both object and source code forms.

IV. AN ALTERNATIVE APPROACH TO COPYRIGHT OF SYSTEMS PROGRAMS

Although the Third and Ninth circuits' opinions were well reasoned and appear consistent with the majority opinion in the CONTU report, their treatment of systems programs did not promote the purpose underlying the copyright laws. The primary purpose of copyright is to ensure "that the welfare of the public will be served and progress of science and useful arts will be promoted by securing to authors for limited periods the exclusive rights to their writings." Rights are granted to authors "primarily for the benefit of the public;" copyright "makes reward to the owner a secondary consideration."

It is difficult to imagine how the public is benefitted by the protection of a systems program that activates the machine or one that checks for memory capacity and then prints the word "Hello." Yet these are some of the systems programs the Ninth and Second Circuits decided to protect in *Apple v. Franklin* and *Apple v. Formula.* In trying to draw a definitive line, the courts may have gone too far. Society is benefitted only slightly when systems programs are protected. In the software industry's current condition, small producers appear to be experiencing some difficulty competing with the major, traditional leaders. Because the small producers must spend time repeating the initial steps, society is deprived of the fruits of their intellectual labor. Society would benefit more if producers were permitted to use established systems programs as a foundation to produce more elaborate programs. Consumers would also benefit because

---

142. Id. See also M. Nimmer, supra note 38, § 1.03[A] (citing Universal City Studios, Inc. v. Sony Corp., 464 U.S. 417 (1984)).
144. See supra notes 104-31 and accompanying text.
145. See supra notes 132-40 and accompanying text.
the systems programs would be designed to operate in more than one company's computer system and, thus, would increase the amount of flexibility allowed to consumers.

Although the value of copyrighting individual systems programs may be questionable, the value increases significantly when the systems program is combined with others or is part of an entire package of software. Determining the limits of copyrightability for systems programs is a difficult, but desirable goal. Although piracy should not be endorsed or encouraged, some flexibility should be allowed so that producers will not have to reinvent the wheel with every project.

One possible solution would be to adopt a method like that suggested by Nimmer in his concurring opinion in the CONTU Report. Alternatively, the courts could reconsider the arguments presented in Franklin that a systems program is a "process," "system," or "method of operation." Perhaps systems programs that merely perform a function within the machine should be considered a process or method of operation, even though such a distinction might prove difficult to draw.

V. PARAMETERS OF COPYRIGHT PROTECTION

Developers and users of software have received substantial guidance both from the statutory provisions for the copyrightability of computer programs and the judicial decisions which have held that programs, whether in object or source code\(^\text{146}\) and regardless of their function, are copyrightable. Nevertheless, some questions remain regarding the practical application of the provisions.

\textbf{A. The Feasibility and Need for Other Methods of Protection}

Over the years, protection of computer programs has assumed various forms. In addition to copyright, legal theories derived from patent and trade secret law have also been used to protect software producers.

\(^{146}\) For an explanation of "source code" and object code, see supra note 86.
1. Patent Protection

Under federal law, patents may protect "any new and useful process, machine, manufacturer, or composition of matter, or any new and useful improvement thereof." A patent may be issued to protect a novel invention, as long as the subject matter is not obvious "to a person having ordinary skill in the art." The patentability of computer programs has been the subject of some dispute.

In *Gottschalk v. Benson*, the Supreme Court held that a program for a digital computer, which converted decimal numbers into pure binary numbers, was not a patentable "process." The Court explicitly stated that the decision did not preclude "a patent for any program servicing a computer," but "the scope of the decision remained far from clear."

The Supreme Court again considered the patentability of computer programs in *Parker v. Flook*. In that case the Court rejected Flook's application to patent a program that updated alarm limits during catalytic conversions because "[t]he only novel feature of the method [was] a mathematical formula." The *Flook* decision "did little, if anything, to settle the issue of whether computer programs were patentable."

The cloud of confusion cleared somewhat in 1981 when the Supreme Court decided *Diamond v. Diehr*. The applicant in *Diehr* sought a patent for a process that collected temperatures, input the data into a computer, and monitored mold presses to ensure the successful curing of synthetic rubber. In a five to four decision, the Court found for the patent applicant even though "in several steps of the process a mathematical equation

148. Id. § 102.
149. Id. § 103.
150. 409 U.S. 63 (1972).
151. Id. at 71.
154. Id. at 585.
157. Id. at 178.
and a programmed digital computer [were] used.” 158 The Court distinguished Diehr from Benson and Flook by noting that Diehr did not seek to patent a mathematical formula. 159 The Court did not determine whether a computer program alone would be patentable, but did state that the use of a computer does not transform an otherwise patentable process into unpatentable subject matter. 160

Although the Supreme Court has apparently cleared the way for patent protection of computer programs, this method of protection is not without its drawbacks. One common complaint is that the patent process is expensive and lengthy: 161 securing a patent may cost as much as 100,000 dollars, 162 and the period between “filing a patent application and issue of a patent typically exceeds two years.” 163 This delay is particularly troublesome for software protection since a program may well be obsolete by the time a patent is issued. 164 In addition, it is difficult to predict whether a particular programmable process is patentable subject matter 165 or to police unlawful use of patents, which are public documents. 166

2. Trade Secret Protection

Because copyright protection is limited to the tangible expression of an idea and does not extend to the underlying idea itself, producers of software often combine copyright protection with trade secret protection. 167 Trade secret law goes further than copyright to protect ideas themselves, regardless of

158. Id. at 185.
159. Id. at 187.
160. Id.
162. Bender, supra note 2, at 418 n. 62 (citation omitted).
163. Id. at 418 (footnote omitted).
164. Posch, supra note 161, at 190.
165. Bender, supra note 2, at 418.
166. Id.
168. See Bender, supra note 2, at 433-38; Conley & Bryan, A Unifying Theory for the Litigation of Computer Software Copyright Cases, 63 N.C.L. Rev. 563, 573-77 (1985); Milgrim, supra note 4, at 158; Posch, supra note 161, at 194. In fact, trade secret protection is the most popular method of protecting software. Bender, supra note 2, at 433 (footnote omitted).
the manner of expression.\textsuperscript{169} According to comment b to section 757 of the Restatement of Torts, 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,"\textsuperscript{170} so long as it remains secret.\textsuperscript{171}

Software is certainly within the scope of trade secret protection, and "the scope of protection seems reasonably clear."\textsuperscript{172} Trade secrets, however, must be kept confidential. To maintain the secrecy necessary for trade secret protection, software producers usually distribute their programs pursuant to licensing agreements that prohibit use or disclosure of the software beyond necessary limits.\textsuperscript{173}

In the past a major problem with the use of trade secrecy to protect computer programs was uncertainty whether the 1976 Copyright Act preempts trade secret protection under state law. Section 301(a) of the Act provides in part:

[All legal or equitable rights that are equivalent to any of the exclusive rights within the general scope of copyright as specified by section 106 in works of authorship that are fixed in a tangible medium of expression and come within the subject matter of copyright as specified by sections 102 and 103, . . . are governed exclusively by this title . . . .] No person is entitled to any such right or equivalent right in any such work under the common law or statutes of any state.\textsuperscript{174}

Section 301 further expresses a legislative intent not to annul or limit "any rights or remedies under the common law or statutes of any state with respect to . . . subject matter that does not

\textsuperscript{170} Restatement of Torts § 757 comment b (1939).
\textsuperscript{171} Id.
\textsuperscript{172} Bender, supra note 2, at 434.
\textsuperscript{173} Conley \& Bryan, supra note 168, at 575. In policing the use of trade secrets, producers should be careful to devote the time and effort needed to protect the secrets. Posch, supra note 161, at 196. One commentator has suggested that the following procedure be used: (1) every document should be marked with a warning notice, (2) circulation of classified documents should be limited, (3) employees should be educated about the importance of maintaining secrecy, (4) employees should sign a nondisclosure agreement, and (5) measures should be adopted to control and restrict access to the documents. Id.
\textsuperscript{174} 17 U.S.C. § 301(a) (1982).
come within the subject matter of copyright as specified in sections 102 and 103; nor to protect "activities violating legal or equitable rights that are not equivalent to any of the exclusive rights within the general scope of copyright as specified by section 106." According to the CONTU report, "The availability of copyright for computer programs does not, of course, affect the availability of trade secrecy protection. Under the Copyright Act of 1976 only those state rights that are equivalent to the exclusive rights granted therein (generally, common law copyright) are preempted." Thus, federal copyright law should not preempt state laws providing trade secret protection.

Trade secret protection provides a number of advantages for software producers. These include the availability of preliminary injunctions, the clear applicability of protection, the wide range of protectible subject matter, the limited opportunity for misuse resulting from restricted circulation of the knowledge, the absence of any waiting period, and the potentially limitless duration of protection. Reliance on trade secret law, however, is not without its disadvantages. First, software producers must bear the burden of developing a trade secret protection system and risk immediate loss of protection should the secret be released. Intellectually, "the necessary fact of secrecy is itself a bar to progress in the sense that when knowledge is not widely proliferated, improvements and further advances based on it are fewer."

Some commentators view trade secret protection as the best method to "protect the program without sacrificing copyright protection or divulging the idea." The advantages of combining the copyright and trade secret protection are twofold. First,

175. Id. § 301(b)(1).
176. Id. § 301(b)(3).
178. D. Bender, supra note 94, § 4A.01[5], at 4A-16.
179. See Posch, supra note 161, at 190; Bender, supra note 2, at 438.
180. D. Bender, supra note 94, § 4A.01[5], at 4A-17.
181. Id.
"protection of the software as a trade secret safeguards the underlying ideas through the use of contractual licensing and related techniques. Layering on copyright protection as well provides protection from third-party recipients who have not signed agreements restricting use or disclosure."\(^\text{183}\) Second, the statutory remedies provided by the copyright law provide "meaningful deterrent[s]."\(^\text{184}\)

**B. The Threat of Infringement and the Burdens of Proof**

The holder of a copyright has the exclusive rights to reproduce, to prepare derivative works, to distribute copies, and to perform and display the copyrighted work.\(^\text{185}\) Infringement occurs when any of the exclusive rights are violated.\(^\text{186}\) To prove infringement, the plaintiff must show (1) the validity of the copyright and (2) the existence of copying.\(^\text{187}\) Under section 410(c) of the 1976 Act, a certificate of copyright registration constitutes "prima facie evidence of the validity of a copyright."\(^\text{188}\) Once the plaintiff has established the prima facie validity of the copyright, the burden shifts to the defendant to overcome the presumption of validity.\(^\text{189}\) The existence of copying may be inferred if the plaintiff shows that the defendant had access to the infringed work and that the original and the allegedly infringing work are substantially similar.\(^\text{190}\)

Showing access to the software is usually not difficult. In

---

183. Milgrim, *supra* note 4, at 163. When combining copyright protection and trade secret protection, "[t]he copyright marking is meant to provide protection only in the event . . . [keeping the program secret] fails." Bender, *supra* note 2, at 436.


186. Id. § 501(a).

187. 3 M. Nimmer, *supra* note 38, § 13.01. See also Novelty Textile Mills, Inc. v. Joan Fabrics Corp., 558 F.2d 1090 (2d Cir. 1977); Sid. & Marty Krofft Television Prods., Inc. v. McDonald's Corp., 562 F.2d 1157 (9th Cir. 1977).


many cases the defendants have admitted access;\textsuperscript{191} in others the
defendants have been licensees or users of the software.\textsuperscript{192} If the
programs are heavily marketed,\textsuperscript{193} the court may presume access
from marketing of the plaintiff’s copyrighted work.\textsuperscript{194} Showing
"substantial similarity" can prove considerably more difficult.

1. Substantial Similarity and Masking

"Substantial similarity" is not easily defined. As Judge
Learned Hand observed, the line establishing substantial simi-
arity is "of necessity vague"\textsuperscript{195} and "wherever it is drawn will
seem arbitrary."\textsuperscript{196} The commonly accepted test of substantial
similarity is the "ordinary observer" test, under which infringement
occurs when "an average lay observer would recognize the
alleged copy as having been appropriated from the copyrighted
work."\textsuperscript{197} The problem with applying this test to computer pro-
grams is that the ordinary person does not have the technical
knowledge to be an informed factfinder. "It is difficult, if not
impossible, to apply these standard copyright principles to a
computer case, primarily because the ordinary lay observer often
is incapable of comparing line by line two sets of computer
source or object code."\textsuperscript{198} To overcome this problem, some pro-
grammers are including unique or identifying language in source
code. The appearance of an original program's unique and
meaningless words in an allegedly infringing work constitutes a

\begin{itemize}
\item \textsuperscript{191} See, e.g., Atari, Inc. v. North Am. Philips Consumer Elecs. Corp., 672 F.2d 607
(7th Cir. 1982); Whelan Assoc. v. Jaslow Dental Lab., Inc., 609 F. Supp. 1307 (E.D. Pa.
\item \textsuperscript{192} In GCA Corp., 1982 Copyright L. Dec. at 17,763, the court found access because the
defendants were former employees of the plaintiff. See also Q-Co Indus., Inc. v.
\item \textsuperscript{193} See, e.g., Apple v. Franklin, supra notes 104-31 and accompanying text.
\item \textsuperscript{194} See, e.g., Midway Mfg. Co. v. American-Bandai, Inc., 546 F. Supp. 125, 146
(D.N.J. 1982).
\item \textsuperscript{195} Peter Pan Fabrics, Inc. v. Martin Weiner Corp., 274 F.2d 487, 489 (2d Cir.
1960).
\item \textsuperscript{196} Nichols v. Universal Pictures Co., 45 F.2d 119, 122 (2d Cir. 1930).
\item \textsuperscript{197} Ideal Toy Corp. v. Fab-Lu Ltd., 360 F.2d 1021, 1022 (2d Cir. 1966). See also
Original Appalachian Artworks, Inc. v. Toy Loft, Inc., 684 F.2d 821, 829 (11th Cir. 1982);
\item \textsuperscript{198} Conley & Bryan, supra note 168, at 582. For an explanation of "source code"
and "object code," see supra note 86.
\end{itemize}
strong indicium of copying. In *Apple Computer v. Franklin*, for example, the name of the Apple programmer was found embedded in the Franklin product.

Even without the use of identifying language, the ordinary lay person will be able to recognize infringement if the infringer makes a verbatim copy. Unfortunately, pirates can employ some simple techniques to mask their infringement. "Procedure and data names may be changed and whole sections of code may be rearranged with just a few keystrokes in a terminal." It is easy, therefore, for a proficient pirate to "render a program virtually unrecognizable in just a few minutes."

Because of the ease of masking programs, industry experts have undertaken to determine what makes a particular program unique. These inquiries have focused on the output, the program’s logic, and the structure of data. In infringement cases the output of the alleged infringing program will usually be similar to the original program’s output. Embodied in output is the program’s unique logic. To determine the logical similarity of two programs, experts generally examine three aspects of the programs: the input, the output, and the processes used to transform the input into the properly formatted output. They may also compare the formatting of the output and input in the two programs.

2. *Substantial Similarity of Programs in Different Languages*

It is particularly difficult to prove similarity of two pro-

199. See, e.g., *Apple v. Franklin*, 714 F.2d at 1245; *SAS Institute*, 605 F. Supp. at 824.

200. 714 F.2d at 1245.


204. "Output" is the tangible hardcopy form of information produced from a computer. See D. Bender, *supra* note 94, § 202.

205. The principles involved are based on the work of Jean-Dominique Warnier and Kenneth T. Orr, who developed the Warnier-Orr Data Structural System Design (DSSD), a technique that allows the building of new software systems. Dakin & Higgins, *supra* note 201, at 134.

206. Id. at 138.
grams that perform the same functions, but are written in different languages. An infringement claim will lie, however, even though the source code or object code of a copyrighted program has been translated into another programming language. A pirating translator faced with to an infringement claim could respond with two arguments.

First, the defendant could argue that since none of the original code was used, the "expression" was not appropriated, and, therefore, no infringement occurred. This argument, however, is easily rebutted. Apple v. Franklin made it clear that "a computer program, whether in object code or source code, is a 'literary work' and is protected from unauthorized copying, whether from its object or source code version."

Second, the translator could argue that the allegedly infringing program itself deserves copyright protection. The pirate could further argue that his program, as "the fruit of intellectual labor," is a work of authorship and that, because it is recorded, it is "expressed in a tangible medium." This argument, however, will also fail. The holder of a copyright has the exclusive right to prepare derivative works. Section 101 of the 1976 Act defines a "derivative work" as "a work based upon one or more pre-existing works." A translation of a literary work, such as a computer program, would be a derivative work since it is based on the preexisting work. Further, the legislative history to section 103 of the Act explicitly states that "an unauthorized translation of a novel would not be copyrighted at all . . . ." Similarly, an unauthorized translation of copyrighted software is not entitled to copyright protection, but is merely an infringing derivative work.

Three recent court decisions have considered the issue of translation piracy and reached similar conclusions. In Whelan

207. See supra note 86.
209. 714 F.2d at 1249.
210. Trade-Mark Cases, 100 U.S. 82 (1879).
212. Id. § 106.
213. Id. § 101.
Associates v. Jaslow Dental Laboratory, Inc., a Pennsylvania federal district court concluded that a program written in BA-
SIC language infringed a copyrighted program that was written
in Event Driven Language (EDL). According to the court, "The expression of the idea embodied in a computer program is
protected by the copyright laws even though it must be altered
and refined to be made adaptable to different types of com-
puters that have different methods of responding to command
controls and therefore require different source codes." The
court found that the defendant had access to the original pro-
gram's source code and that the similarity between the two pro-
grams was extensive. In addition to accepting expert testi-
mony to evaluate the degree of similarity between the two pro-
grams, the court considered the "almost identical" visual
screens that were displayed by the programs and accepted testi-
mony that prospective users and buyers could not find substan-
tial differences between the two programs.

In SAS Institute, Inc. v. S & H Computer Systems, Inc., a Vanderbitl University professor required the use of SAS, the
plaintiff's sophisticated statistical computing program. A prob-
lem arose, however, because Vanderbilt had a DEC VAX com-
puter system and SAS was available only for IBM and IBM
compatible computers. To solve the problem, the professor,
some of his colleagues, and S & H formed a limited partnership
to develop an SAS program that would operate on a DEC VAX
system. Toward this end, the defendant became a licensee of
the SAS program and obtained the SAS source code. SAS In-
stitute alleged that S & H, in the process of preparing its prod-
uct, made unauthorized copies of the SAS source code and that
the S & H product was either a copy or a derivative work of

216. EDL is a recognized computer language. Id. at 1310. See supra note 86.
217. Id. at 1320 (citation omitted).
218. Id. at 1321. In finding similarity, it appears that the court relied heavily on
expert testimony and was inclined to rule in plaintiff's favor because the defendant's
expert was not well informed about the programs at issue. See id. at 1321-22.
219. Id. at 1322.
220. 605 F. Supp. 816 (M.D. Tenn. 1985). For a discussion of the case by one of the
attorneys involved, see Conley & Bryan, supra note 168, at 582 & n. 158.
221. 605 F. Supp. at 819.
222. Id. at 821.
SAS. 223

The court found for the plaintiff on both claims. Considering the second infringement claim, the court stated that the defendant had duplicated the plaintiff's expression, not just its ideas, and that a substantial similarity existed between the organization and structure of the defendant's and the plaintiff's programs. 224 The court also found that because S & H targeted SAS for duplication, the S & H program was an unauthorized derivative work within the statutory definition. 225

Unlike the defendants in Whelan and SAS Institute, the defendant in E. F. Johnson Co. v. Uniden Corp. of America 226 did not have access to the source code. E. F. Johnson (EFJ) developed a technologically advanced mobile radio and repeater system and a compatible mobile radio. 227 In an effort to develop a competitive mobile radio that was compatible with the EFJ radio and repeater system, Uniden accessed the EFJ program and created its own software. Uniden admitted that engineers removed the EFJ microchip, which stored the object code of the program, and converted the object code into a programming language from which Uniden flowcharted the EFJ program. 228 Because of the similarities in the EFJ and Uniden programs, EFJ claimed copyright infringement and moved for a preliminary injunction.

The court considered the plaintiff's likelihood of success on the merits and found prima facie evidence of a valid copyright. 229 The court also found that Uniden had access to the program and that substantial similarity existed between the two programs. The court cited similarity in two programs' logic as evidence of substantial similarity and noted that the Uniden

223. Id. at 828.
224. Id. at 830. The trial court found at least forty-four specific examples of copying. Id.
225. Id. at 831.
227. Id. at 1487. When EFJ first conceived the idea for its trunk radio system, there were no trunk radio systems on the market. Id. at 1488. According to the court, "'[T]runking' of frequency channels permits the system to afford all system users automatic access to all channels for maximum efficiency." Id. at 1489.
228. Id. at 1490.
229. The court concluded that the copyright registration was prima facie evidence of the copyright. Id. at 1492 (citing 17 U.S.C. § 410(c) (1982); Novelty Textile Mills, Inc. v. Joan Fabrics Corp., 558 F.2d 1090, 1091 n. 1 (2d Cir. 1977)).
program did not employ the most efficient steps to perform certain functions. In addition, the Uniden program contained unnecessary instructions that had been included by mistake in the EFJ program, and the court concluded "that Uniden unwittingly copied the unnecessary instructions when translating the EFJ code virtually verbatim."

Although the programs did not match up line for line when converted to the same programming language, the court followed Whelan and found that "line-by-line comparison becomes meaningless" when examining a converted program. The court granted the plaintiff's motion for a preliminary injunction and held that the defendant could develop software to compete with the plaintiff's, but not "by copying the exact number and sequence of bytes or items by which Plaintiff's program causes the machine to operate."

C. Defenses

1. Reverse Engineering

Regardless of the uniqueness of a computer program, the copyright protection extends only to the expression and not to the underlying idea. The use of ideas is unrestricted. Thus, the door is always open for a programmer to learn the logic of a program and then rewrite it in a noninfringing manner. Unfortunately, "No principle can be stated as to when an imitator has gone beyond the 'idea,' and has borrowed its 'expression.'" The CONTU report considered the process of reverse engineering and found that "[w]hen other language is available, programmers are free to read copyrighted programs and use the
ideas embodied in them in preparing their own works."\textsuperscript{238} The report also noted that "one is always free to make the machine do the same thing as it would if it had the copyright work placed in it, but by one's own creative effort rather than by piracy."\textsuperscript{239} The second work, as the "fruit of intellectual labor,"\textsuperscript{240} would also be eligible for copyright protection.

On the other hand, reverse engineering may thwart the policy underlying copyright protection. One of the purposes of copyright is to "persuade authors to make their ideas freely accessible to the public so that they may be used for the intellectual advancement of mankind."\textsuperscript{241} If imitation is permitted, software developers will resort increasingly to trade secret protection,\textsuperscript{242} and the free access to ideas will be stifled. The producer of software is understandably fearful of marketing its software only to watch a competitor market a similar version a few months later, either by piracy or by "legitimate" reverse engineering.

When a party asserts the defense of "reverse engineering," the court must analyze the similarities between the two programs very carefully. Since it was the idea that the second program borrowed from the original, the logical similarity between the two will necessarily be "very close."\textsuperscript{243} In addition, the sequence of events in a reengineered program will probably also be similar to the original. Computer programs produced through reverse engineering do not necessarily infringe, but if the reengineered program follows the original too closely, an infringement may occur. The burden rests upon the reengineering party to prove reverse engineering, and that burden should be a difficult one. Otherwise common piracy could be erroneously classified as reverse engineering. When a defendant claims reverse engineering, a court should analyze the similarities of the programs even more than in other alleged infringement cases.

\textsuperscript{238} CONTU Final Report, supra note 46, at 20 (citations omitted).
\textsuperscript{239} Id. at 21.
\textsuperscript{240} Trade-Mark Cases, 100 U.S. 82 (1879).
\textsuperscript{241} 3 M. Nimmer, supra note 38, § 13.03[A] (citing Eichel v. Martin, 241 F. 404, 410 (S.D.N.Y. 1917)).
\textsuperscript{242} See supra notes 167-73 and accompanying text.
\textsuperscript{243} Dakins & Higgins, supra note 201, at 140.
2. Fair Use

"Fair use," a common defense to infringement claims, is rarely raised in software cases. The fair use doctrine, derived from section 107 of the Act and considered "an equitable rule of reason," defies attempts at specific definition: "each case . . . must be decided on its own facts." Nevertheless, the statute delineates the following four factors that should be considered when determining whether a specific use of a work is a fair use:

(1) the purpose and character of the use, including whether such use is of a commercial nature or is for nonprofit educational purposes; (2) the nature of the copyrighted work; (3) the amount and substantiality of the portion used in relation to the copyrighted work as a whole; and (4) the effect of the use upon the potential market for or value of the copyrighted work.

Although both user and competitor pirates could assert a fair use defense, the results in the two cases would differ considerably because of the different degree of commercial use.

a. User Piracy

In *Sony Corp. v. Universal City Studios, Inc.*, the Supreme Court considered whether copying television programs for later viewing was a legitimate fair use and decided that it was. The Court's application of the four statutory fair use factors, however, suggests some differences between "timeshifting" and user reproduction of software.

The Court adopted the lower court's conclusion regarding the first factor and found that "time-shifting for private home use must be characterized as a noncommercial, nonprofit activity." Similarly, the purpose and character of a user pirate's copying of a work could be classified as noncommercial, nonprofit activity since the user pirate normally copies a program stored on a disk for his own convenience rather than for later

---

247. Id. at 449.
Discussing the second factor, "the nature of the copyrighted work," the Sony court concluded that "time shifting merely enables a viewer to see such a work which he had been invited to witness in its entirety free of charge."248 In contrast, the programs copied by most user pirates are not offered free of charge; the user must purchase them. In addition, the software producer intends to sell additional copies of the program as demand rises. Thus, when a user buys one copy and reproduces it as the need increases, the copying conflicts with the purpose or nature of the producer's work. A user could argue, however, that since section 107 provides an exception for certain types of copying, copying is not completely contrary to the nature of the work. Thus, non-profit reproduction, at least for in-house use, should not be prohibited.

A court basing its fairness assessment heavily on the third statutory factor, the amount of the program copied, would probably find that copying by a noncommercial consumer is not a fair use. A user would almost always copy the entire program since a single portion would be useless. The Sony Court, however, did not rely on this factor in its analysis. In fact, the Court explicitly stated that "the fact that the entire work is reproduced . . . does not have its ordinary effect of militating against a finding of fair use."249 If courts adopt a similar attitude toward user piracy, placing less emphasis on the portion copied, this factor should not prove fatal to a defense of fair use.

The decisive factor for the Sony Court appears to have been the fourth one, the effect of the use on the market.250 According to the Court, "A challenge to a noncommercial use of a copyrighted work requires proof either that the particular use is harmful, or that if it should become widespread, it would adversely affect the potential market for the copyrighted work."251 Although unauthorized copying of software could reach significant proportions and hurt the profits from software sales, it is uncertain whether the original software producer's market would be drastically reduced by customers copying software instead of

248. Id.
249. Id. at 449-50 (citation omitted).
250. Id. at 450-51.
251. Id. at 451.
buying additional packages. The misuse would have to be so rampant that it noticeably affected a company's profits.

Thus, a user pirate facing a claim of copyright infringement would be well advised to raise the defense of fair use. The argument might fail because of the potential harm to the original software producer's market; on the other hand, the uncertainty of the harm might induce a court to accept the defense.

b. Competitor Piracy

The competitor pirate would have great difficulty in supporting a defense of fair use. When the use has a commercial or profit-making purpose, a presumption of unfair use arises. In *Harper & Row Publishers, Inc. v. Nation Enterprises*, the Supreme Court identified the crucial issue as "whether the user stands to profit from exploitation of the copyrighted material without paying the customary price." A competitor of an original software producer who pirates the original producer's software obviously stands to profit from the copying; the pirate can avoid a substantial investment in development and then pass the savings on to the customer.

Application of the second and third statutory factors to a competitor piracy case would yield the same result as in a user piracy case. The fourth factor, however, would probably be fatal to the competitor pirate's defense. The dramatic distinction between user and competitor pirates lies in the effect each produces on the software market. When the copying is for commercial gain, as in the case of competitor piracy, a showing of actual present harm is not needed, and the likelihood of future harm may be presumed. Thus, the competitor pirate would have a heavy burden to overcome in proving fair use.

VI. ExtraJudicial Solutions to Piracy

Perhaps the most encouraging recent development for the software industry is the realization by computer manufacturers

---

252. *Id.* at 449.
254. 105 S. Ct. at 2231-32 (citations omitted).
255. 464 U.S. at 451.
that cooperation with each other can reduce the likelihood of some piracy. IBM, for example, recently announced a plan to join forces with Microsoft Corporation, a software supplier, to develop operating systems programs\textsuperscript{256} for personal computers. Under their agreement, Microsoft will have the right to market the systems programs to other computer manufacturers. Software producers have expressed relief that IBM is not planning to exclude the rest of the industry from producing software to run on future IBM computers.\textsuperscript{257} If such access had been available to Franklin Computer, infringement might still have occurred, but would have made Franklin's defense of alleged good faith efforts to produce Apple compatible software untenable.\textsuperscript{258} If the SAS system in \textit{SAS Institute} had been available for the DEC VAX Vanderbilt system, the professor could have gained the benefits legitimately, without infringing.\textsuperscript{259} The IBM-Microsoft agreement should promote compatibility and encourage competition. With greater compatibility, computer owners will be able to choose from a wider range of software producers and will not be obliged to purchase products only from the hardware manufacturer.

Promotion of hardware compatibility was the motivation behind formation of the Corporation for Open Systems, an industry association that is attempting to develop standards for enabling computers manufactured by different companies to communicate with each other.\textsuperscript{260} Despite this and similar efforts, however, infringement will undoubtedly continue at least until all hardware and software products are interchangeable, and probably even beyond. The stringent law being developed in the courts will help protect the original producer's exclusive rights.

\textsuperscript{256} For an explanation of "operating systems programs," see \textit{supra} note 105.


\textsuperscript{258} See \textit{supra} notes 132-40 and accompanying text.

\textsuperscript{259} See \textit{supra} notes 220-25 and accompanying text.

\textsuperscript{260} \textit{Computer Makers Seeking Standards}, N.Y. Times, Jan. 6, 1986, at D4, col. 3. The companies participating in the organization include the Burroughs Corp., Digital Equip. Corp., AT&T, Control Data, Hewlett-Packard, Honeywell, NCR, Sperry, Telex, Wang Laboratories, and Xerox. IBM is noticeably absent from the list of current members, but may join later. \textit{Id.}
VII. CONCLUSION

Software producers have genuine fears about their ability to market a unique product that will not be pirated by competitors or users. Statutory provisions and case law indicate that application and systems programs, whether of object code or source code, are copyrightable. Recent decisions, however, may have gone too far in holding that all systems programs are protectible. To show infringement of a copyrighted program, a plaintiff must demonstrate access to the copyrighted work as well as substantial similarity. In software cases courts should be especially careful to analyze the programs for substantial similarities despite the difficulties in comparing source codes.

Two affirmative defenses are available for software pirates. Reverse engineering should protect competitor copiers in a proper case. The "fair use" defense, on the other hand, is more likely to aid user pirates than competitor pirates.

Finally, society may look to the industry to circumvent the need to copy by cooperating to make software and hardware universally compatible. These efforts should be welcome news both for computer purchasers and for the courts. Consumers will have more, and cheaper, options when shopping for hardware and software. The courts, on the other hand, will see fewer infringement cases, and in those that do arise, the parties' arguments should be easier to evaluate.

Sylvia Ann Matthews