1-1-1992

Shared Rights to Source Code: The Copyright Dilemma

Alexandra J. Horne

Follow this and additional works at: http://digitalcommons.law.scu.edu/lawreview

Part of the Law Commons

Recommended Citation

Available at: http://digitalcommons.law.scu.edu/lawreview/vol32/iss2/4

This Comment is brought to you for free and open access by the Journals at Santa Clara Law Digital Commons. It has been accepted for inclusion in Santa Clara Law Review by an authorized administrator of Santa Clara Law Digital Commons. For more information, please contact sculawlibrarian@gmail.com.
COMMENTS

SHARED RIGHTS TO SOURCE CODE: THE COPYRIGHT DILEMMA*

I. INTRODUCTION

Over the last two hundred years copyright law in the United States has changed dramatically to meet the demands of creative and technological development. Yet, copyright law has continued to reflect the intent of the framers of the Constitution, who gave Congress the power "to promote the progress of science and useful arts, by securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries." In the last twenty-five years, with the rapid growth of computer technology, this constitutional mandate has become increasingly difficult to fulfill and has necessitated many changes in copyright law through congressional legislation and judicial interpretation.

One area of intellectual property protection that has

* ©1992 by Alexandra J. Horne. The author wishes to thank June Carbone, Jeffrey Steinhardt, and Joel Linzner for their helpful suggestions, and especially her husband, Joe, for his encouragement, patience and unflagging sense of humor.

1. Copyright law is defined as "a system of property rights for certain kinds of intangible products, generally called works of authorship." MARSHALL LEAFFER, UNDERSTANDING COPYRIGHT LAW 1 (1989). It developed in 18th century England to give authors the exclusive right to make copies of their books, and thus, refers to the right to make copies. Id. Today, copyright protection is much broader, covering most artistic, literary, and musical works, as well as computer software. Id. See infra notes 24-62 and accompanying text.

2. U.S. CONST. art. I, § 8, cl. 8. Following the American Revolution, the thirteen colonies, with the exception of Delaware, passed laws protecting authors; but because these laws only applied within the boundaries of each state, problems arose that required a uniform national law. The result was Article I, Section 8, Clause 8 of the Constitution. Id. at 4-5.

3. See infra notes 24-153 and accompanying text.

4. Intellectual property law concerns the protection of property rights to information. LEAFFER, supra note 1, at 14. For example, copyright protects expressive information, patent protects technological information, and trademark protects symbolic information. LEAFFER, supra note 1, at 14. Although these three forms of
been particularly challenging is the protection of the rights to computer software. Software copyright is an area that developed very rapidly and often did not fit comfortably with earlier copyright law. In fact, it was unclear whether software could be protected by copyright until 1980 when the Copyright Act of 1976 was amended, following the recommendations of the National Commission on New Technological Uses of Copyrighted Works (CONTU).

Copyright protection of source code and object code has been an especially troublesome area of software protection, as has the protection of application and operating system protection are based on different theories and statutes, they often overlap and interrelate, due to the nature of the rights and the fact that the federal government administers much of intellectual property law, with the exception of trade secret protection, which is governed by state law. LEAFFER, supra note 1, at 14. See infra notes 225-34 and accompanying text for a discussion of trade secret and patent protection.

5. Software tells the computer how to accomplish functions that produce a specific result or perform a task, such as screen displays for a video game, columns in an accounting program spreadsheet, or “spell check” in a word processing program. Id. at 66. “Software is the instructions to the computer.” ALAN FREEDMAN, THE COMPUTER GLOSSARY 636 (1989). A series of instructions that performs a specific task is called a software program or a program. Id. The Copyright Act of 1976 defines a computer program (software) as “a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result.” 17 U.S.C. § 117 (1988).

6. See infra note 64.

7. Act of December 12, 1980, Pub. L. No. 96-517 § 10, 94 Stat. 3015, 3028 (codified as amended at 17 U.S.C. §§ 101, 117 (1988)). The idea of creating CONTU began in the 1960’s, but the Commission was not established until 1974 with the passage of an amendment to the Copyright Act. Pub. L. No. 93-573 § 201, 88 Stat. 1873, 1873-74 (1974). Title II of that amendment stated that the purpose of CONTU was to study the “use of copyrighted works of authorship . . . in conjunction with automatic systems capable of storing, processing, retrieving, and transferring information . . . and the creation of new works by the application or intervention of such automatic systems . . . .” Id. at 1873. The Commission was directed to make recommendations “as to such changes in copyright law or procedures that may be necessary to assure for such purposes access to copyrighted works, and to provide recognition of the rights of copyright owners.” Id. at 1873-74. See infra notes 48-54 and accompanying text.

8. Source code is a “high level” programming language, such as BASIC, FORTRAN, or PASCAL, that uses English symbols and is understandable to a human. LEAFFER, supra note 1, at 66.

9. Object code is the machine executable instructions created by a compiler or assembler from source code. BRYAN PFAFFENBERGER, QUE’S COMPUTER USER’S DICTIONARY 324 (1990).

10. Most litigation concerning the copyrightability of computer programs does not involve source code. ALAN LATMAN ET AL., COPYRIGHT FOR THE NINETIES 170 (1989). This is because source code is not physically incorporated into the com-
tem programs,11 and the literal and nonliteral elements of a computer program.12 The protection of the rights to computer source code is of particular concern to a software developer, because to be economically viable, a software developer must be able to develop and sell software, but retain the rights to portions of the source code. Otherwise, the developer would have to create new subroutines13 each time new application software is developed.14

A conflict as to ownership of source code can arise when a software publisher, for whom a software package has been developed, claims exclusive rights to the entire software package through a licensing agreement and subsequent copyright protection.15 This exclusivity would preclude reuse of the source code by the developer and require starting from scratch each time a new software package is developed for a different publisher.16 Clearly this control over the software rights would...
have a severe economic impact on the developer, which, in turn, would be passed on to the public in the form of higher prices and fewer products: results that could conflict with the constitutional mandate to benefit society." Conversely, if the software developer retains the rights to the source code, not only does the developer receive an economic incentive to continue exploration of new technological innovations, but the benefits are shared by the publisher and the user in the form of product variety, efficiency and price.

This comment addresses the issue of shared rights to source code, focusing on copyright protection as the preferred method of protection. Beginning with section II, the comment sets the stage with a history of copyright law in the United States and how it has been adapted and amended over the years to accommodate the evolving needs of the computer industry and the constantly changing age of high technology. This section also discusses a number of cases that laid the groundwork for later developments in this field. Section III explores the problem of shared rights to source code in the context of a recent dispute in the Ninth Circuit, and section IV analyzes the problem in terms of established copyright law and its development, comparing software protection to that of literary characters. Finally, section V proposes a practical solution for copyright protection of source code in today's era of rapidly developing software. The proposal suggests an amendment to copyright law that will strengthen statutory protection of software and reflect the prevailing custom in the

copyright in our law." H.R. REP. NO. 1476, 94th Cong., 2d Sess. 123 (1976). Specifically, section 101 defines a copyright owner "with respect to any one of the exclusive rights comprised in a copyright . . . ." 17 U.S.C. § 101 (1988). However, divisibility does not mean that multiple owners can share the rights to portions of a software package or the underlying source code, as this comment will explain. Rather, section 101 refers to the divisibility of the rights enumerated in section 106, such as reproduction, derivative works, distribution of copies, or public performance or display. 17 U.S.C. § 106 (1988). See Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) ¶ 26,612 (N.D. Cal. 1990).

17. See infra notes 219-21 and accompanying text.
18. Trade secret and patent protection will also be considered as alternative forms of protection. See infra notes 225-34 and accompanying text.
19. See infra notes 24-54 and accompanying text.
20. See infra notes 55-153 and accompanying text.
21. See infra notes 154-69 and accompanying text.
22. See infra notes 170-234 and accompanying text.
23. See infra notes 235-44 and accompanying text.
software industry today. Alternatively, the proposal recommends guidelines for the courts to follow when interpreting existing copyright law.

II. BACKGROUND

A. Evolution of Copyright Protection

Copyright law has been a vital force in the United States since the framers of the Constitution gave Congress the power to establish copyright protection. In response to this mandate, and to protect the needs of the arts and sciences in a developing nation, Congress passed the first Copyright Act on May 31, 1790. Although there is no legislative history for the Act, it is clear that Congress used the British Statute of Anne as its guide. The Copyright Act of 1790 protected the rights of an author to any book, map or chart for a term of fourteen years, renewable for an additional fourteen years.

Between 1790 and 1909, when the Copyright Act was revised, Congress addressed the developments in technology and the arts by gradually adding more subjects that could be protected by copyright. For example, in 1802, prints were added; in 1831, musical compositions; during the Civil War, photographs; and in 1870, paintings, drawings, and sculpture.

24. U.S. CONST. art. I, § 8, cl. 8. Because this clause was adopted in final form without debate in a secret proceeding on September 5, 1787, little is known about what the framers had in mind or the scope of the language itself.


26. 8 Anne ch. 19 (1710). The Statute of Anne was passed in England in 1710. It was the first law to recognize the rights of authors and has been used throughout the western world as an example for copyright law.


1. 1909 Copyright Act

In 1909, the Copyright Act was revised pursuant to a 1905 request by President Theodore Roosevelt, who wanted to make the law more compatible with the times. This new Act of 1909 expanded copyrightable subject matter to cover "all the writings of an author." In addition, it increased the duration of copyright to an original term of 28 years and a renewal term of twenty-eight years, doubling the length of possible copyright protection. This Act remained in effect for sixty-seven years until Congress completely overhauled it with the Copyright Act of 1976.

2. 1976 Copyright Act

In the intervening years, from 1909 to 1976, additions to copyrightable subject matter addressed changes in technology, such as the invention of motion pictures and sound recordings. It was not until 1955, however, that Congress finally authorized funding for a study by the Copyright Office to determine how best to meet the needs and anticipate the difficulties that a new, revised copyright act would raise. In response, the Copyright Office prepared a number of reports regarding the potential problems in any major revision of the law and made recommendations as to how best to revise the law.

35. Id. § 24 (current version at 17 U.S.C. § 302 (1988)).
These recommendations were given to Congress in 1961.\textsuperscript{41} Many years of Senate and House hearings followed\textsuperscript{42} until finally, the current Copyright Act was passed in 1976.\textsuperscript{43} The new law made numerous changes to existing copyright law. It established a single federal system of copyright protection and expressly preempted state common law.\textsuperscript{44} It also provided protection for “original works of authorship fixed in any tangible medium of expression, now known or later developed, from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device.”\textsuperscript{45} Copyrightable subject matter was again expanded to include seven broad categories: literary works; musical works; dramatic works; pantomimes and choreographic works; pictorial, graphic, and sculptural works; motion pictures and other audiovisual works; and sound recordings.\textsuperscript{46} Additionally, the

\begin{thebibliography}
41. Id.
44. 17 U.S.C. § 301 (1988). Section 301(a) provides:
On and after January 1, 1978, 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, whether created before or after that date and whether published or unpublished, are governed exclusively by this title. Thereafter, no person is entitled to any such right or equivalent right in any such work under the common law or statutes of any State.
Id. § 301(a). Section 301(b) specifies exceptions to this rule, such as subject matter that does not come under copyright protection, any cause of action that began prior to January 1, 1978, and activities that violate legal or equitable rights that are not within the general scope of copyright. Id. § 301(b). Section 301(c) concerns sound recordings, id. § 301(c), and section 301(d) states that “[n]othing in this title annuls or limits any rights or remedies under any other Federal statute.” Id. § 301(d).
45. Id. § 102(a). Note that this section overrules an early Supreme Court decision that a player piano roll did not qualify for copyright protection because it was not in a form intelligible to a human. White-Smith Music Publishing Co. v. Apollo Co., 209 U.S. 1 (1908). Today, sound recordings, computer programs, and motion pictures are examples of works that cannot be read without the aid of a machine or device, and yet, are copyrightable. LEAFFER, supra note 1, at 92.
\end{thebibliography}
Act simplified the duration of copyright protection to a term equal to the life of the author plus fifty years.47

3. CONTU Recommendations

Although the Copyright Act of 1976 incorporated many changes, it did not address protection of computer technology, delegating that question to the National Commission on New Technological Uses of Copyrighted Works (CONTU) for resolution.48 While Congress awaited CONTU’s recommendations, section 117 of the Act, entitled “Limitations on exclusive rights: Computer programs,” left protection of computer programs as it had been under the 1909 Copyright Act. This section provided the owner of a copyright with no “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.”49

With the establishment of CONTU in 1976, the age of computers and copyright protection finally came into focus. CONTU’s charter directed it to make recommendations to Congress concerning how copyright law could best address these new challenges.50 CONTU released its final report on July 31, 1978, with the following recommendations regarding

47. Id. § 302(a). Protection for joint works is for a term of the life of the last surviving author plus 50 years. Id. § 302(b) (1988). Protection for anonymous works, pseudonymous works or works for hire is for a term of 75 years from the date of first publication or 100 years from the year of creation, whichever expires first. Id. § 302(c).
49. Id. Section 117 in the 1976 Copyright Act specified in full:
   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.
   Id.
computer software:

The new copyright law should be amended 1) to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright; 2) to apply to all computer uses of copyrighted programs by the deletion of the present Section 117; and 3) to assure that rightful possessors of copies of computer programs can use or adapt these copies for their use.\[51\]

Congress adopted these recommendations on December 12, 1980, with an amendment to the Copyright Act.\[52\] Specifically, section 101 of the 1976 Copyright Act was revised to include a definition of computer programs as “a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result.”\[53\] In addition, section 117, which concerned the limitations to exclusive rights with regard to computer programs, was revised to incorporate CONTU’s recommendations.\[54\]

B. Copyright Protection of Computer Software

Today, computers are an integral part of personal and business life, because they are able to perform complex calculations many times faster than the human brain. To perform these functions, the computer uses a software program.\[55\] A computer software program is developed by a human being (known as a computer programmer) using source code. Although source code is understandable to a human, it is not intelligible to a computer. Therefore, in a simplified case, a two-step process is used to translate the source code into a machine-readable, binary language of ones and zeros.\[56\] In the

---

51. COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS (CONTU), FINAL REPORT 2 (1978) [hereinafter FINAL REPORT].
53. Id. § 10(a).
54. Id. This section deals explicitly with situations in which copies or adaptations of computer programs are not an infringement. Id. Specifically, a copy or adaptation can be made if it is an “essential step in the utilization of the computer program” or if it is used for archival purposes. Id.
55. LATMAN ET AL., supra note 10, at 161. See supra note 5 and accompanying text.
56. Binary language is a series of ones and zeros, representing electrical
first step, a computer program, called a compiler,\textsuperscript{57} translates the source code into an intermediate language. Next, a second computer program, an assembler,\textsuperscript{58} translates the intermediate language into object code, a binary machine language that the computer understands and can execute directly.\textsuperscript{59} This final version is stored in the computer's memory in the central processing unit (CPU) or on floppy diskettes,\textsuperscript{60} where it can be accessed by the computer user.\textsuperscript{61}

Although the Copyright Office registered the first computer program in 1964,\textsuperscript{62} the passage of the Act of December 12, 1980 gave the judiciary a solid foundation on which to base decisions concerning the copyrightability of computer programs. One of the first cases to interpret this new legislation, and a landmark decision concerning copyright protection of computer software, was \textit{Apple Computer v. Franklin Computer Corp.}\textsuperscript{63}

\begin{itemize}
\item \textsuperscript{57} A compiler translates a "high level" language, such as BASIC, COBOL, FORTRAN, or PASCAL, into machine-readable code. MEADOWS ET AL., \textit{DICTIONARY OF COMPUTING \& INFORMATION TECHNOLOGY} 155 (1987). It is "the computer's native language" and "the only language the computer understands." FREEDMAN, \textit{supra} note 5, at 354.
\item \textsuperscript{58} A "floppy" is "an auxiliary memory device consisting of a flexible magnetic disk resembling a phonograph record, which can be inserted into the computer and from which data or instructions can be read." Apple Computer v. Franklin Computer Corp., 714 F.2d 1240, 1243 (3d Cir. 1983).
\item \textsuperscript{59} A "machine-readable language or "machine language" is a "very explicit set of instructions" that the computer is able to understand and execute. \textit{ENCyclopedia Of COMPUTER SCIENCE \& ENGINEERING} 1885 (A. Ralston \& E.D. Reilly, Jr. eds. 1983).
\item \textsuperscript{60} In 1964, the Copyright Office began to accept registration of computer programs under its "rule of doubt," but required that the deposit be in a form that a human could read (e.g., a printout) if the program had been published in a form that only a machine could read. LEAFFER, \textit{supra} note 1, at 64 n.25.
\end{itemize}
1. **Apple Computer v. Franklin Computer Corp.**

In 1983, when *Apple v. Franklin* was decided, the question of copyright protection of computer software was very unsettled. In fact, the district court, in denying Apple's motion for a preliminary injunction, stated that it had "some doubt as to the copyrightability of the [computer] programs" in general. However, in reversing the lower court's decision, the appellate court noted that "[its] legal ruling [was] fundamental to all future proceedings in this action, and . . . ha[d] considerable significance to the computer services industry."

Apple Computer was and is a leader in the field of personal computers and related software; Franklin, a much smaller competitor. From the start of the case, Franklin admitted that it had copied Apple's operating system program in order to manufacture a computer that was compatible to an Apple computer, but defended its copying because "it was not feasible for Franklin to write its own operating system programs." Franklin's vice president of engineering testified

---

64. One of the problems with copyrightability of software and computer programs was that they have a different "look and feel" from traditional forms of copyrightable subject matter. Leaffer, *supra* note 1, at 65. Even members of CONTU had doubts, arguing that other forms of copyrightable subject matter communicate with people, while computer programs are "essentially mechanical, labor-saving devices, which transmit electronic impulses to operate machines called computers." Leaffer, *supra* note 1, at 65. (quoting Dissent of Commissioner Hershey, *Final Report*, *supra* note 51, at 27-28).


66. 714 F.2d at 1242. Apple originally filed suit against Franklin in 1982, alleging that Franklin was liable not only for copyright infringement of 14 computer programs, but for patent infringement, unfair competition and misappropriation, as well. Franklin responded to the charge of copyright infringement with the affirmative defense that the computer programs did not contain copyrightable subject matter and made a counterclaim for a declaratory judgment that the copyright registrations were invalid and unenforceable. Following discovery, Apple moved for a preliminary injunction, which the lower court denied, resulting in this appeal. *Id.* at 1244-45.

67. *Id.* at 1242-43. At the time of the appeal, Apple manufactured Apple II computers and distributed over 150 software programs. In 1981, it had sold over 400,000 Apple II computers, had approximately 3,000 employees and annual sales of $335 million. Franklin, on the other hand, manufactured the ACE 100 personal computer, and at the time of the appeal, had 75 employees and had sold less than 1,000 computers. *Id.*

68. *Id.* at 1245.
that he had studied the requirements and concluded that development was not feasible, because it was too technically challenging. 69 However, he stated that Franklin had only attempted to rewrite one Apple program and admitted that additional programs probably could have been rewritten, a premise that was confirmed by Apple's evidence that others had in fact written Apple-compatible programs. 70

On appeal, the court addressed three major issues concerning the copyrightability of computer programs: (1) whether a computer program expressed in object code, as distinguished from source code, is copyrightable; 71 (2) whether a computer program embedded in Read Only Memory (ROM) 72 is copyrightable; 73 and (3) whether a computer operating system is copyrightable. 74

a. Copyrightability of Object Code

The court began its discussion of the copyrightability of object code with a statement from its decision in Williams Electronics v. Artic International 75 that "the copyrightability of computer programs is firmly established after the 1980 amendment to the Copyright Act." 76 In Williams, the court rejected the defendant's argument that source code, which the defendant thought copyrightable, should be distinguished from object code, which the defendant thought was not copyrightable. 77 Expanding on that holding, the court determined that object code was copyrightable, even though it must be deciphered by

---

69. Id. Specifically, David McWherter testified that "there were just too many entry points in relationship to the number of instructions in the program." Id. Entry points are used by programmers to "mesh" application programs with operating system programs, and McWherter determined that identical signals would have to be used for 100% compatibility. Id.

70. Id.

71. Id. at 1246. See infra notes 75-79 and accompanying text.

72. Read Only Memory (ROM) is an "internal permanent memory device consisting of a semi-conductor or "chip" which is incorporated into the circuitry of the computer . . . . Information stored on a ROM can only be read, not erased or rewritten." 714 F.2d at 1243. In that respect, it is similar to a compact disk, video disk or phonograph record. FREEDMAN, supra note 5, at 581.

73. 714 F.2d at 1246. See infra notes 80-81 and accompanying text.

74. 714 F.2d at 1246. See infra notes 82-99 and accompanying text.

75. 685 F.2d 870 (3d Cir. 1982).

76. 714 F.2d at 1248 (quoting Williams Elecs. v. Artic Int'l, 685 F.2d at 875).

77. 685 F.2d at 876.
a machine, because the Copyright Act of 1976 extended copyright protection to "works in any tangible means of expression 'from which they can be perceived, reproduced, or otherwise communicated directly or with the aid of a machine or device.'" In addition, the court confirmed that a computer program, in either object or source code, could be classified as a literary work for copyright protection.

b. Copyrightability of Computer Program Embedded in ROM

Secondly, the court rejected the defendant's argument that ROMs are not copyrightable because they are "utilitarian objects or machine parts." Again, citing its recent decision in Williams, the Apple court stated that the "embodiment of the expression in ROM devices" establishes fixation and thus, is copyrightable.

c. Copyrightability of Operating System Program

Finally, the court turned to what is considered the most difficult question: the copyrightability of operating system programs as distinguished from application programs. Franklin's position was that pursuant to section 102(b) of the 1976 Copyright Act, operating system programs were per se uncopyrightable. In addition, the defendant argued that, as

---

79. 714 F.2d at 1249. When computer programs are created, the human programmer uses source code, but for the computer to understand the program, it must be translated into object code, a binary machine language, which the computer can execute directly. See supra notes 55-61 and accompanying text. Because object code is very difficult for a human to interpret, and thus, difficult to copy, it is the form that is usually registered with the Copyright Office and distributed publicly. LEAFFER, supra note 1, at 66.
80. 714 F.2d at 1249.
81. Id. (citing Williams Elecs. v. Artic Int'l, 685 F.2d 870 (3d Cir. 1982)).
82. Application programs generally perform specific tasks for the computer user, such as word processing or bookkeeping. Operating system programs, on the other hand, are transparent to the user and perform internal computer functions or facilitate the use of application programs. 714 F.2d at 1242-43.
83. Section 102(b) distinguishes copyright protection from patent protection, stating that "[i]n 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) (1988).
set forth in *Baker v. Selden*, an operating system program was "either a 'process', [sic] 'system', [sic] or 'method of operation' and hence uncopyrightable." The court noted that section 102(b) and *Baker v. Selden* both distinguish between patent law, which protects discoveries, and copyright law, which protects writings. However, Franklin's application in this case was erroneous, because Apple intended to copyright the instructions, rather than the method for using the instructions. Additionally, the court determined that there was no difference between the protection of instructions for application programs and system programs, because both tell the computer to perform a specific operation and should be given the same copyright protection under section 102(b).

The court also disagreed with the lower court's position that an operating system program was part of a machine, stating that simply because an operating system program "may be etched on a ROM does not make the program either a machine, part of a machine or its equivalent." In fact, an operating system can be removed from the computer if it is resident on a diskette or tape, and according to CONTU, "should no more be considered machine parts than videotapes should be considered parts of projectors or phonorecords parts of sound reproduction equipment."

---

84. 101 U.S. 99 (1879). At issue in *Baker v. Selden* was the contention that the ruled lines and headings used to illustrate the bookkeeping system were part of the copyrighted book, and therefore, "no one [could] make or use similar ruled lines and headings, or ruled lines and headings made and arranged on substantially the same system, without violating the copyright." *Id.* at 101. The court disagreed with this position, finding:

The description of the art in the book, though entitled to the benefit of copyright, lays no foundation for an exclusive claim to the art itself. The object of the one is explanation; the object of the other is use. The former may be secured by copyright. The latter can only be secured, if it can be secured at all, by letters patent.

*Id.* at 105. Thus, the court concluded that "blank account-books are not the subject of copyright; and . . . the mere copyright of Selden's book did not confer upon him the exclusive right to make and use account-books, ruled and arranged as designated by him and described and illustrated in said book." *Id.* at 107. See also infra notes 105-08 and accompanying text.

85. 714 F.2d at 1250.

86. *Id.* at 1251.

87. *Id.*

88. *Id.*

89. *Id.* (quoting FINAL REPORT, supra note 51, at 21).
Franklin argued that because the operating systems were "purely utilitarian works," they were not copyrightable pursuant to *Baker v. Selden.*\(^90\) The court stated that Franklin's interpretation of that decision was too broad and cited the more recent Supreme Court decision in *Mazer v. Stein,\(^91\) which rejected such an expansive view.\(^92\) As additional support, the court quoted the CONTU majority: "[T]he words of a program . . . used ultimately in the implementation of a process should in no way affect their copyrightability,"\(^93\) and noted that Congress had adopted CONTU's recommendations almost verbatim.\(^94\)

The court stated that the strongest argument in support of the copyrightability of both operating system and application programs was that the statutory definition of computer programs did not distinguish between them.\(^95\) Furthermore, Franklin was unable to cite any case law to support its distinction between the two types of programs.\(^96\)

Finally, Franklin claimed that because only a limited number of ways existed to achieve compatibility with the Apple computer, the idea and expression merged, thereby precluding copyrightability.\(^97\) The court rejected this argument, explain-
ing that if there were other ways to express an idea, there was no merger. Thus, the court rejected Franklin's argument that operating system programs were per se uncopyrightable and reversed the lower court's decision, stating that it could find nothing from either Congress or CONTU that distinguished between the copyrightability of an operating system program and an application program.

Clearly, this decision established the copyrightability of computer programs and created a solid foundation for future questions of copyright protection and infringement, such as Whelan Associates v. Jaslow Dental Laboratory.

2. Whelan Associates v. Jaslow Dental Laboratory

In 1986, in Whelan Associates v. Jaslow Dental Laboratory, the Third Circuit extended copyrightability to nonliteral elements of a computer program. At issue was whether "the structure (or sequence and organization) of a computer program [was] protectible by copyright, or whether the protection of the copyright law extends only as far as the literal computer code." Noting that Apple Computer v. Franklin Computer Corp. had established that both source code and object code (the literal elements of the program) were copyrightable, the Whelan court discussed the protection of the nonliteral elements or structure of the program, beginning with an analysis of Baker v. Selden.

Baker v. Selden is the seminal case for distinguishing between an idea, which is not protectible under copyright law,
and an expression, which is protectible. In Baker, the Supreme Court determined that blank forms, which were an indispensable part of Baker's accounting method, were part of the idea, and thus, not copyrightable. "In other words, the purpose or function of a utilitarian work would be the work's idea, and everything that is not necessary to that purpose or function would be a part of the expression of the idea."

In Whelan, the court determined that the structure of the program was part of the expression, not the idea. The court based its decision on the fact that the CONTU Final Report indicated that the Commission intended copyright protection to extend beyond the literal code, and, more importantly, that copyright protection beyond the literal code to the structure and the logic would provide the "proper incentive" to the computer programmer. The court noted that coding is a relatively small part of programming in comparison to the amount of effort that must be put into the overall development of a computer program. In fact, it is much more expensive and difficult to develop the structure and logic of the program, including debugging, documentation and logic, than it is to do the actual coding. Thus, to protect the high cost of program development, the court determined that copyright protection should extend beyond the literal code and encompass the program's structure, sequence and organization.

106. 797 F.2d at 1234. See supra note 84.
108. 797 F.2d at 1236.
109. Id. Since this comment first went to the publisher, the Second Circuit has criticized this position as an "extremely broad view of copyrightability for a computer program," using "what now seems to be a simplistic test for similarity between computer programs" that is both "inadequate and inaccurate." Computer Assocs. Int'l v. Altai, Inc., 775 F. Supp. 544, 558-59 (2d Cir. 1991).
110. Id. at 1241 (quoting FINAL REPORT, supra note 51, at 21). The Report stated that "the separation of idea from form of expression . . . is better realized through the courts exercising their judgment in particular cases [than by a per se rule] . . . . Flow charts, source codes, and object codes are works of authorship in which copyright subsists." Id. (emphasis omitted). Source code and object code are examples of the literal elements of a computer program, and the program's structure, sequence and organization are examples of nonliteral elements. Id. at 1248.
111. 797 F.2d at 1237.
112. Id. at 1231.
113. Id.
114. Id. at 1248. But see Plains Cotton Coop. v. Goodpasture Computer Serv.,
Once a court has determined that an expression is copyrightable, it must decide whether the copyright has been infringed.\textsuperscript{113} Given the nature of litigation, the defendant generally does not admit to copying, which would constitute direct evidence, and the court must rely on circumstantial evidence for its decision.\textsuperscript{116} Using a two-part test, the court determines first, whether the defendant had access to the copied element, and if so, whether there is substantial similarity between the plaintiff's product and the defendant's allegedly infringing product.\textsuperscript{117}

In the first part of the test, after access and similarities have been shown, the plaintiff must demonstrate that the similarities are sufficient to prove copying.\textsuperscript{118} To prove these similarities, the plaintiff can use expert testimony to assist the trier of fact in resolving the issue.\textsuperscript{119} Once copying has been shown, the next step is to prove unlawful appropriation or illicit copying.\textsuperscript{120} In this second part of the test, expert testimony is not allowed, and the finder of fact must rely solely on the "ordinary lay hearer."\textsuperscript{121}

\textsuperscript{807} F.2d 1256 (5th Cir. 1987), in which the court did not protect the structure, sequence and organization of the computer program, finding that the numerous structural similarities were "compelled" by the nature of the cotton market. Although this decision appears on the surface to reject Whelan, the preliminary injunction may have been denied solely because of an insufficient showing of alternative ways to structure the allegedly infringing program. LATMAN ET AL., supra note 10, at 460.

Subject to certain, narrow limitations, section 106 of the Copyright Act provides the owner of a copyright with the exclusive rights to that work. 17 U.S.C. § 106 (1988). Consequently, anyone who violates that right "infringes" the copyright, and the copyright owner is entitled to institute an action for infringement. \textit{Id.} § 501.

Note that the defendants' admissions of copying in both Apple Computer v. Franklin Computer Corp., 714 F.2d 1240, 1245 (3d Cir. 1983), and Lotus Development Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 68 (D. Mass. 1990), are the exceptions, rather than the rule.

Arnstein v. Porter, 154 F.2d 464 (2d Cir. 1946). This case concerned a claim that Cole Porter had infringed a number of the plaintiff's musical compositions. In response to this charge, the court established the bifurcated test for copyright infringement which is still in general use today. \textit{Id.} Note that on remand, the court found for the defendant. \textit{See} Arnstein v. Porter, 158 F.2d 795 (2d Cir. 1946).

\textsuperscript{118} 154 F.2d at 468.

\textsuperscript{119} \textit{Id.}

\textsuperscript{120} \textit{Id.}

\textsuperscript{121} \textit{Id.}
In Whelan, the district court questioned the use of this "bifurcated substantial similarity test" when the subject matter was highly complex and unfamiliar to most lay observers, such as the structure of computer software programs. The appellate court agreed with the lower court's use of a non-bifurcated test, stating that when the subject matter of the copyright was especially complex, the ordinary lay observer test was only marginally useful and potentially misleading.

3. *Lotus Development Corp. v. Paperback Software International*

Recently, the decision in Whelan was adopted by the Second Circuit in *Lotus Development Corp. v. Paperback Software International*. This case concerned the copyrightability of nonliteral elements of expression in the user interface of Lotus' 1-2-3 computer spread sheet program. As in Whelan,
structure, sequence and organization of the computer program were considered, as well as the menu command system.\textsuperscript{127}

The court first examined whether Congress had extended copyright protection to the nonliteral elements of computer programs and, if so, to what extent.\textsuperscript{128} After considering the relevant language of the Copyright Act, the court decided that “[a]lthough Congress did not include ‘computer programs’ in [the] list of examples of ‘works of authorship,’ [that] computer programs [fell] squarely within the statutory definition of literary works.”\textsuperscript{129} The court noted, however, that limitations apply to copyright protection and “[i]n 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.”\textsuperscript{130}

As additional support, the court quoted a congressional report, which stated that “[s]ection 102(b) [was] intended . . . to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of copyright law.”\textsuperscript{131} The court therefore determined that Congress intended copyright protection to extend to “original expression embodied in computer programs,” a conclusion consistent with the treatment of nonliteral elements of expression in literature, musicals, plays and motion pictures.\textsuperscript{132}

\textsuperscript{127} Id. at 82-83.
\textsuperscript{128} A menu command system is a screen display that presents a list of commands with a “moving cursor,” which is used to select a particular command, thereby instructing the computer to perform a specific function. Id. at 63-64.
\textsuperscript{129} Id. at 46-51.
\textsuperscript{129} Id. at 49 (citing 17 U.S.C. § 101 (1988)). Section 101 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.” 17 U.S.C. § 101 (1988).
\textsuperscript{130} 740 F. Supp. at 49 (quoting 17 U.S.C. § 102(b) (1988)).
\textsuperscript{132} 740 F. Supp. at 54. The court noted that even though an infringer did not copy the words or dialogue of a book, play or musical score, the work still
Next, the court considered the defendant's contention that the user interface in Lotus 1-2-3 was a useful article and as such, not entitled to copyright protection.\textsuperscript{13} The court determined that simply because the computer program was useful or functional did not preclude its being copyrightable and pointed out that one statutory mandate should not "destroy or absorb" another.\textsuperscript{14} Indeed, the court emphasized that any other position would "deny copyright protection to the most original and least obvious products of the creative mind" simply because they were functional, and would result in the marketing of inferior products for fear that the best products might be too good for copyright protection.\textsuperscript{15}

Finally, the court discussed the copyrightability of the nonliteral elements of the program, using a three-part test: (1) to identify the idea and distinguish it from its expression; (2) to determine if the expression was limited to essential elements of the idea (merger); and (3) to decide if the expression was a substantial part of the copyrightable work.\textsuperscript{16} Noting the "amorphous nature of nonliteral elements of computer programs,"\textsuperscript{17} the court decided that some of these elements were not copyrightable, such as the two-line moving cursor\textsuperscript{18} and the rotated "L,"\textsuperscript{19} because of the limited number of ways that these ideas could be expressed.\textsuperscript{20} However, the court de-

could infringe if the setting, characters, or plot were copied and resulted in substantial similarity of expression. \textit{Id.} at 51 (citing Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930)). As Judge Learned Hand stated in \textit{Nichols}, "the right cannot be limited literally to the text, else a plagiarist would escape by immaterial variations." 45 F.2d at 121.

133. 740 F. Supp. at 52.

134. \textit{Id.} at 57-58. In other words, elements of expression are copyrightable even though functionality or usefulness cannot be used as the basis for copyright protection. \textit{Id.}

135. \textit{Id.} at 58.

136. \textit{Id.} at 60-62.

137. \textit{Id.} at 46.

138. \textit{See supra} note 127.

139. An electronic spreadsheet is much the same as a manual spreadsheet, in that it "presents a blank form on which numerical, statistical, financial or other data can be assimilated, organized, manipulated and calculated." 740 F. Supp. at 68. According to the court, the spreadsheet screen display representing Lotus' spreadsheet resembled "an 'L' rotated ninety degrees clockwise with letters across the top to designate columns, and numbers down the left side to designate rows." \textit{Id.}

140. \textit{Id.} at 65. \textit{See supra} notes 97-98 and accompanying text for a discussion of the merger doctrine.
terminated that taken as a whole, the program's user interface was copyrightable, because the menu command structure was not essential to the idea of an electronic spreadsheet and could be expressed in many ways.\footnote{740 F. Supp. at 68. The court relied heavily on Judge Learned Hand's "Abstraction Test" in Nichols v. Universal Pictures Corp., 45 F.2d 119 (2d Cir. 1930), to distinguish between the idea and expression, although it stated that it was not necessary (or always even possible) to "completely disentangle" the idea from its expression. \textit{Id.} at 60. In Nichols, Judge Hand stated:  

\textit{ Upon any work, and especially upon a play, a great number of patterns of increasing generality will fit equally well, as more and more of the incident is left out. The last may perhaps be no more than the most general statement of what the play is about, and at times might consist only of its title; but there is a point in this series of abstractions where they are no longer protected, since otherwise the playwright could prevent the use of his "ideas," to which, apart from their expression, his property is never extended. Nobody has ever been able to fix that boundary, and nobody ever can.} \textit{45 F.2d at 121.}}


Currently, the law concerning shared ownership of source code is unsettled. One of the few federal cases to deal with this issue is \textit{Digital Communications Associates v. Softklone Distributing Corp.}\footnote{659 F. Supp. 449 (N.D. Ga. 1987).} In this 1987 case, the court held that screen displays could be protected independently from the underlying object code or source code.\footnote{Digital Communications Assocs. v. Softklone Distrib. Corp., 659 F. Supp. 449 (N.D. Ga. 1987).} However, since that decision, Congress has determined that copyright protection of a computer program extends to the screen displays, and that infringement of either the computer program or the screen display is an infringement of the other.\footnote{37 C.F.R. § 202.3(b) (1990). \textit{See also} 53 Fed. Reg. 21,817 (1988).} Thus, the Copyright Office now accepts a single registration covering all copyrightable elements of expression in a computer program,\footnote{LEAFFR, supra note 1, at 188-89 n.41.} making it unnecessary to copyright the screen displays and the underlying code separately.

Although it may no longer be necessary to follow the precedent set in \textit{Softklone}, the court's discussion of the idea/expression dichotomy continues to have impact as a basic
theme throughout all areas of copyright protection. The court began its discussion with a statement from Whelan that "'[i]t is axiomatic that copyright protects only the expression of ideas and not the ideas themselves.'"\(^{146}\) This protection was codified in 17 U.S.C. 102(b) and "'[i]n no case does copyright protection for an original work of authorship extend to any idea . . . regardless of the form in which it is described, explained, illustrated, or embodied in such work.'"\(^{147}\)

Congress has discussed this distinction with particular application to computer programs, because of the concern that copyright of computer programs should only protect the expression of the programmers' ideas and not extend to methods or processes.\(^{148}\) To that end, one of the purposes of section 102(b) was to clarify the distinction between idea and expression: the programmer's expression was copyrightable, but the "actual processes or methods embodied in the program" were not.\(^{149}\)

Thus, as Softklone and the previous three decisions demonstrate, the judiciary has looked favorably on the expansion of copyright protection for computer software. In Apple v. Franklin, the court extended copyright protection to object code and operating system programs.\(^{150}\) In Whelan, copyrightability was expanded to cover the nonliteral elements of the computer program,\(^{151}\) a decision reaffirmed by Lotus.\(^{152}\) Finally, the discussion of the idea/expression dichotomy in Softklone emphasized one of the most fundamental tenets of copyright law: an expression of an idea is copyrightable, but the idea itself, is not.\(^{153}\)

III. THE PROBLEM

Although software is clearly copyrightable, a conflict can occur when two parties disagree as to its ownership, and unfor-

---

146. 659 F. Supp. at 457 (citing Whelan Assocs. v. Jaslow Dental Lab., 797 F.2d 1222, 1234 (3d Cir. 1986)).
149. Id.
150. See supra notes 64-101 and accompanying text.
151. See supra notes 102-24 and accompanying text.
152. See supra notes 125-41 and accompanying text.
153. See supra notes 142-49 and accompanying text.
fortunately, a licensing agreement does not always answer that question satisfactorily. In Accolade, Inc. v. Distinctive Software, Inc., for example, Distinctive Software, Inc. (DSI) entered into a licensing agreement to develop software games with Accolade, Inc., a software publisher that sells video games worldwide. According to Accolade, DSI "expressly agreed that . . . Accolade would be the owner of the copyright for all products, or derivative works from those products, which were developed by DSI pursuant to the License Agreement." In fulfilling this agreement, DSI developed a number of games for Accolade, including a very popular driving game: The Duel.

In its complaint, Accolade alleged that it owned the copyright to The Duel and that DSI had created and developed a second driving game that was substantially similar to The Duel in the use of its underlying source code. In addition, Accolade alleged that DSI had copied subroutines from The Duel and that by doing so, DSI had infringed Accolade's copyright in The Duel. Accolade therefore asked for a preliminary injunction to prevent any further infringement of its copyright.

In DSI's response to copyright infringement, DSI agreed that it had entered into a development contract with Accolade in October 1987, but stated that during the negotiations leading up to the contractual agreement, DSI had "insisted that it retain ownership of its 'library routines' and development tools." In addition, DSI stated that not only was Accolade

157. Id. at 1.
158. Id. at 4-5.
159. Id. at 1.
160. Id. at 2.
161. Id.
162. Id.
163. Defendant's Memorandum in Opposition to Plaintiff's Motion for Prelimi-
aware that DSI used these library routines to develop all of its games, but that it was the custom in the software industry for the developer to retain the rights to generic source code subroutines to be reused in subsequent development efforts.\textsuperscript{164}

DSI proposed that the court interpret the contract "in accordance with industry custom and practice," arguing that industry experts confirmed that it was the custom for a software developer to retain the rights to the library routines or functional routines, unless there was explicit language to the contrary.\textsuperscript{165} Additionally, DSI claimed that the custom and practice were so pervasive in the software industry, that "a judgment granting software publishers a copyright over their developers' tools and routines could cause the collapse of the computer game software industry."\textsuperscript{166}

DSI claimed that knowing these facts, Accolade had agreed to allow DSI to retain the source code as a trade secret, and had embodied this agreement in the October 1987 contract.\textsuperscript{167} Therefore, DSI claimed it had not infringed Accolade's copyright, because the copyright did not apply to the source code.\textsuperscript{168} Conversely, the plaintiff interpreted the licensing agreement to expressly provide Accolade with the "exclusive ownership of a valid copyright to all portions of The Duel," a fact that Accolade says DSI acknowledged by placing a notice of Accolade's copyright on all copies of the game.\textsuperscript{169}

\textsuperscript{164} Id. at 2.
\textsuperscript{165} Id. at 13.
\textsuperscript{166} Id.
\textsuperscript{167} Id. at 4.
\textsuperscript{168} Id. at 5.
\textsuperscript{169} Plaintiff's Memorandum of Points and Authorities in Support of Motion for Preliminary Injunction at 6-7, Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) ¶ 26,612 (N.D. Cal. 1990) (No. 90-20202).
IV. ANALYSIS

As evidenced by Accolade, a problem can arise as to the ownership of the copyright to source code, even though both parties appear to understand the terms of the agreement at the time the contract is signed. To protect the rights to the underlying source code library routines, the software developer should turn to established copyright law, beginning with an argument used by authors to retain the rights to their characters in literary works.170

A. Protection of Characters


At first glance it might seem that a thirty-eight-year-old case concerning the protection of fictional characters would have no application to computer program copyrightability; but in fact, Warner Bros. Pictures v. Columbia Broadcasting System171 established an argument that can be used to protect source code today. In Warner Bros., the plaintiff claimed copyright to the detective story, “The Maltese Falcon,” including its characters and their names. Subsequently, the author, Dashiell Hammett, reused the characters from “The Maltese Falcon” in new stories that were broadcast by Columbia Broadcasting System (CBS) as the “Adventures of Sam Spade,” prompting Warner Brothers to sue for copyright infringement.172

In its decision, the court first addressed the construction of the contract, noting that the use of characters and their names had not been explicitly mentioned in the contractual agreement assigning the rights to the “Maltese Falcon” to Warner Brothers.173 Quoting Philipp v. Jerome H. Remick & Co.,174 the court stated that any doubt as to ownership should be resolved in favor of the author and that the “clearest lan-

170. Computer programs have been copyrighted as literary works, pursuant to 17 U.S.C. § 102 (1988). Therefore, the arguments protecting characters in literary works are easily applied to computer programs.
171. 216 F.2d 945 (9th Cir. 1954).
173. Id. at 949.
guage is necessary to divest the author of the fruits of his labor. Such language is lacking here." Because the characters were not even mentioned in the agreement, the court determined that neither the rights to the characters nor their names had been granted to Warner Brothers.

The court found historical as well as current support for its decision in the tradition that detective writers carried forward the leading characters from one story to the next. In addition, the court noted that Warner Brothers had not objected to Hammett's use of the characters in three stories prior to this suit and concluded that the parties had intended that "Hammett should [not] be deprived of using the Falcon characters in subsequently written stories, and that the contract, properly construed, [did] not deprive Hammett of their use." This determination was supported by Warner Brothers' purchase price, which was inadequate to include the rights to such popular literary characters.

In its search for explicit statutory support, the court explained that although the practice of writing sequels was old and the Copyright Act had been amended several times, the Act had never specifically addressed this question; nor had the question ever been adjudicated, although it was mentioned in Nichols v. Universal Pictures Corp. Thus, the court determined that "[i]f Congress had intended that the "sale of the right to publish a copyrighted story would foreclose the author's use of its characters in subsequent works ... Congress would have made specific provisions therefor."

The court supplemented this position with a discussion of the limitations of an author's imagination, commenting that like an artist, an author's characters are "always limited

---

175. 216 F.2d at 949 (quoting Philipp v. Jerome H. Remick & Co., 145 F. Supp. 756 (S.D.N.Y. 1936)).
176. 216 F.2d at 949.
177. Id. The court cited earlier authors such as Edgar Allen Poe and Sir Arthur Conan Doyle and current detective writers such as S. Van Dine and Erle Stanley Gardner to support its position. Id.
178. Id. at 950.
179. Id. at 949.
180. Id. at 950 (9th Cir. 1954) (citing Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930)).
181. 216 F.2d at 950.
and... fall into limited patterns."\textsuperscript{182} For this reason, the court concluded that it would be unreasonable to restrict the reuse of characters by their author and would in fact, be contrary to the Copyright Act's purpose to promote the arts.\textsuperscript{183} In addition, the court decided that the characters were the vehicle for the story and as such, did not go with the sale of the story.\textsuperscript{184} However, although the court determined here that the characters were not copyrightable and that the author was free to use them in subsequent stories, it did not foreclose the possibility that characters could be copyrightable in the future.\textsuperscript{185}


As established in *Warner Bros. Pictures v. Columbia Broadcasting System*,\textsuperscript{186} the court should first look to the language of the contractual agreement to see if any express language assigned the rights to the entire body of source code to the software publisher. Next, it should consider historical and/or current custom in the industry, as well as the intent of the parties at the time of the agreement. Finally, the court should evaluate the concept of limited creativity.

In *Accolade, Inc. v. Distinctive Software, Inc.*,\textsuperscript{187} for example, the licensing agreement stated: ""Each licensed product shall be delivered to Publisher in the form of a duplicatable program (a "Master") on tape, diskette, and/or other specified medium .... In addition to the Master, certain other materials delivered concurrently shall be prepared by Developer. These deliverables shall include ... source code.""\textsuperscript{188} As previ-

\textsuperscript{182} Id.
\textsuperscript{183} Id.
\textsuperscript{184} Id.
\textsuperscript{185} Id. Indeed, the Ninth Circuit has found that characters are copyrightable when they ""really constitute"" the story and are not merely vehicles of the story. See *Walt Disney Prods. v. Air Pirates*, 581 F.2d 751 (9th Cir. 1978). Another theory concerning the copyrightability of literary characters was put forth by Judge Learned Hand in *Nichols v. Universal Pictures Corp.*, 45 F.2d 119 (2d Cir. 1930). Judge Hand stated that ""the less developed the characters, the less they can be copyrighted; that is the penalty an author must bear for marking them too indistinctly."" Id. at 121. In other words, characters must be sufficiently delineated before copyright protection can attach.

\textsuperscript{186} 216 F.2d 945 (9th Cir. 1954).
\textsuperscript{187} Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) \textsuperscript{188} 26,612 (N.D. Cal. 1990).

\textsuperscript{188} Id. at 23,625-26 (quoting Defendant's Exhibit 7, at 8, Memorandum in
ousely discussed, Accolade interpreted the agreement as granting it the copyright to the source code in its entirety, while DSI contended that the copyright protection did not include the source code library routines.189

DSI argued that rather, the development agreement specified that the source code should be delivered in addition to the developed and licensed product (The Duel), and therefore, the copyright protected only the licensed product, which did not include the source code.190 The district court agreed with DSI's interpretation and denied Accolade's motion for a preliminary injunction against the developer.191 Although the court did not feel that ownership of the copyright had been fully briefed or argued for a final disposition, its reading of the language of the licensing agreement at the time of its order was that it did not transfer ownership of the underlying source code to Accolade.192

Next, the court should look to historical and/or current custom in the industry to see whether the rights to portions of the source code may be retained by the developer. In Accolade, DSI claimed that it was the custom in the computer industry for the developer to retain the rights to its library routines for reuse in subsequent development efforts.193 To support its claim, DSI provided the court with numerous declarations by experts in the field of computer programming and software development, affirming this tradition.

189. See supra notes 155-69 and accompanying text.
191. Id. at 23,628.
192. Id. at 23,626-27.
194. “It is an accepted fact in the industry that a developer could not continue in operation if it transferred its ability to use its developed library of standard functional tools to a particular publisher for whom it was doing work.” Declaration of Emil Heidkamp in Opposition to Plaintiff's Request for Preliminary Injunction at 4, Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) ¶ 26,612 (N.D. Cal. 1990) (No. 90-20202).

It is the custom and practice in our industry that when a copyright in a computer game is conveyed to a publisher, the property rights
In addition to industry practice, the court should consider the intent of the parties at the time of the agreement, whether the plaintiff has objected to any previous use of the source code, and the price paid for the licensing agreement. In Accolade, DSI claimed that it had always insisted on retaining the rights to its library routines, offering a statement by the former president of Accolade that during negotiations:

Accolade initially insisted upon the transfer to Accolade of all bytes to the entire source code for each computer game developed by DSI for Accolade, [but] DSI vigorously resisted that demand and it insisted that it retain ownership of the source code. *Accolade gave considerable thought to this issue and decided to allow DSI to retain the source code.*

In addition, DSI claimed that Accolade consistently "acted in a manner contrary to the notion that it owned the rights to DSI's library routines." For example, Accolade offered to

transferred are limited to the unique aspects of the computer game as expressed in the video display and all rights to the library of generic or functional tools remain with the developer for use in future computer game development projects.


"[I]t is universally understood within the entertainment software publishing business that this library of basic, low level code or functional subroutines can and will be reused by the developers in performing subsequent development contracts with other publishers . . . ." Declaration of Robert Lindsey in Opposition to Plaintiff's Motion for Preliminary Injunction at 3, Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) ¶ 26,612 (N.D. Cal. 1990) (No. 90-20202).

It is commonly understood by publishers working in the industry that the well-established independent software developers they contract with often have developed—and have used in the development of programs for the various publishers they work for—a library of common, often repeated routines that are used as tools or building blocks in virtually every specific game program they build.

Declaration of Don Daglow in Opposition to Plaintiff's Motion for Preliminary Injunction at 3, Accolade, Inc. v. Distinctive Software, Inc., Copyright L. Dec. (CCH) ¶ 26,612 (N.D. Cal. 1990) (No. 90-20202).


196. Defendant's Memorandum in Opposition to Plaintiff's Motion for Prelimi-
buy DSI's library routines, but DSI refused and instead, provided technical support to help Accolade develop its own library of subroutines. Furthermore, Accolade paid DSI for the development of many additional products, even though DSI had never delivered the source code for the library routines, indicating that Accolade considered DSI's work to be complete without delivery of that source code. Finally, as in Warner Bros. Pictures v. Columbia Broadcasting System, DSI claimed that Accolade was aware that DSI had used the library routines in its development for third parties and, until this suit, had been indifferent to that use. In fact, even upon actual notice of the use, Accolade had never previously claimed copyright infringement.

Finally, the concept of limited creativity, set forth in Warner Bros., has application to software development. As previously discussed, a software developer generally spends a great deal of time determining the most effective means to achieve a specific result. Under such exhaustive circumstances, the developer who has determined the most efficient configuration of source code does not want to repeat the extensive development cycle for each new contract. Indeed, an argument can well be made that a software developer, like a literary author, is an artist with a finite number of ideas.

A software developer could implement this argument of limited creative possibilities in support of copyright protection of source code. In a situation such as Warner Bros. or Accolade, when a conflict arises after the parties have signed the contract, the software developer could argue that there are only a limited number of ways to write the source code for the same subroutines. Because they "fall into limited patterns," the developer should be able to retain the rights to the library routines to be protected by a separate copyright or as a trade secret. However, the ownership of the delivered product, developed especially to meet the unique needs of each individual

---


197. Id. at 12-13.

198. Id.

199. See supra notes 111-14 and accompanying text.

publisher, should belong to that software publisher. This argument is similar to the merger doctrine, where an idea and expression merge, but here, the limitations are of a creative nature and do not preclude copyright protection of the underlying source code.

B. Copyright Protection

If a developer decides to use copyright to protect the rights to source code library routines, the first step is to register the source code with the Copyright Office at the time the library routines are developed. Indeed, the developer might prefer copyright rather than trade secret as a means of protection, because it could provide a stronger defense against future publishers’ claims of copyright infringement.

In order for a work of authorship to be copyrightable, it must be original to that author and meet a very minimal standard of creativity. Accordingly, once an original work is created, “[c]opyright protection subsists . . . in original works of authorship fixed in any tangible medium of expression.”

Although passage of the 1976 Copyright Act made registration with the Copyright Office optional, because registration is prima facie evidence of ownership of a valid copyright and a prerequisite for filing any suit for infringement, it is an option most would favor when choosing copyright protection.


203. See infra notes 225-29 and accompanying text.

204. 17 U.S.C. § 102(a) (1988). It is very rare that works do not meet the requisite minimum level of creativity for copyright protection. For cases in which the court found lack of creativity, see John Muller & Co. v. New York Arrows Soccer Team, 802 F.2d 989 (8th Cir. 1986); Magic Mktg. v. Mailing Servs. of Pittsburgh, 694 F. Supp. 769 (W.D. Pa. 1986).


206. Id. § 408(a).

207. Id. § 411(a).
With these options in mind, the prudent software developer would want to register the source code that performs basic library functions with the Copyright Office prior to entering into any development agreements with a software publisher. In this way, the developer would protect the rights to these subroutines for reuse as needed in future development agreements, as well as protect against the tedious, time-consuming task of creating new solutions to the same problem for each new development agreement.

In *Accolade, Inc. v. Distinctive Software, Inc.*, for example, DSI developed its library of subroutines long before entering into a development agreement with Accolade. In fact, much of the code was developed by an individual prior to joining DSI. Certainly, when the code has been developed over a period of time and used in multiple development agreements, the resulting library routines are more likely the property of the developer, rather than the publisher. Indeed, DSI asserted that it owned prior copyrights to the underlying source code.

1. *Economic Support*

As discussed earlier, copyright protection is the most prudent means to protect the rights to previously developed source code, a proposition that is supported by *Whelan Associates v. Jaslow Dental Laboratory*. Although the conflict in *Whelan* was dissimilar to a software developer/publisher relationship, the court's emphasis on the amount of work necessary to develop a software program has application to all types

---

209. *Id.* When Rick Friesen joined DSI as Director of Research and Development, he brought his library of subroutines to be used by DSI in its development efforts. *Id.*
211. 797 F.2d 1222 (3d Cir. 1986).
of software development. Although the court found that the "coding process [was] a comparatively small part of programming," its discussion of the effort and expense needed prior to final coding supports any economic argument made by a developer for copyright protection of source code.

In describing the development process, the court noted that first the programmer must identify the problem and outline a solution in terms of software modules or subroutines. Next, the programmer arranges the subroutines to maximize the efficiency of the program, a critical factor in the development of a useful program. Only when the detailed design of the program is complete does the programmer begin the actual coding. Thus, the Whelan court is correct in stating that "[b]y far the larger portion of the expense and difficulty in creating computer programs is attributable to the development of the structure and logic of the program, and to debugging, documentation and maintenance, rather than to the coding."

It is also correct to assume that once a developer has expended the extensive time and effort needed before writing the source code, there is a strong economic incentive to retain the exclusive rights to that code, even though the actual writing is not particularly demanding. Once the most efficient manner of accomplishing the desired result is developed, the developer has a vested interest in protecting it against any infringing uses. Using copyright, the developer is able to protect against any conflict of interest in subsequent development contracts and the necessity of creating new source code to satisfy identical functional requirements. Indeed, it is this very premise that formed the backdrop for the court's decision in Whelan.

212. See supra notes 111-14 and accompanying text.
213. Whelan Assocs. v. Jaslow Dental Lab., 797 F.2d at 1231.
214. Id. at 1229-30.
215. Id. at 1230.
216. Id.
217. Id. at 1231.
218. Id. Note that the decision in Whelan concerned the copyrightability of nonliteral elements of the computer program, while the issue here concerns only the literal elements of the developer's source code. Nevertheless, the discussion of the creative process has application to the copyrightability of both literal and nonliteral elements of computer programs.
2. Legislative and Judicial Support

In addition, the Constitution and Congress bring strong economic support to the argument for separate copyright protection of computer source code subroutines. As previously discussed, the copyright statutes are based on Article I of the Constitution, which gives Congress the power "to promote the progress of science and useful arts, by securing for limited times to authors and inventors the exclusive right to their respective writings and discoveries." Certainly the promotion of the useful arts entails economic considerations and incentives, a position that has found favor with the legislature over the years.

In the legislative report on the 1909 Copyright Act, for example, Congress discussed the need to balance the benefit to the public, with the benefit to the author. More recently, Congress has recognized that "[a]lthough the primary purpose of the copyright law is to foster the creation and dissemination of intellectual works for the public welfare, it also has an important secondary purpose: To give authors the reward due them for their contribution to society.

The legislative intent that the authors of copyrighted works reap economic benefit has found complementary support with the judiciary. In Mazer v. Stein, for example, the Supreme Court discussed the economic philosophy behind the constitutional grant as the "conviction that encouragement of individual effort by personal gain is the best way to advance public welfare."

Thus, the software developer, whose concerns are economic viability and potential growth, can find strong support from many sources for the proposition that it should not be

223. Mazer v. Stein, 347 U.S. at 219 (1954). This sentiment was echoed in Sony Corp. of America v. Universal City Studios, 464 U.S. 417 (1984), in which the Court stated that the limited monopoly of copyright protection was intended to "motivate the creative activity of authors . . . by the provision of a special reward." Id. at 429.
necessary to recreate library routines each time the developer enters into a new agreement with a software publisher. Indeed, the developer finds support across the spectrum of legal authority: from the legislature, the judiciary, and historically, from the Constitution. Furthermore, because of the complexity of the development process, the software developer can argue that the library routines must be afforded copyright protection because of the economic hardship the developer will face due to the prohibitive expense required to “reinvent the wheel” for each new development project.  

C. Trade Secret and Patent Protection

In addition to the federal statutes, the developer can turn to state law or common law for trade secret protection. To receive trade secret protection, the developer must maintain the confidentiality as to the way in which the source code is written, and in return, the courts will protect against the misappropriation of that code. Obvious problems can arise, however, with regard to maintaining the secret, both intra-organizationally and particularly when the developed product has wide distribution. Therefore, trade secret protection is more valuable during product development, where

224. Defendant’s Memorandum in Opposition to Plaintiff’s Motion for Preliminary Injunction at 13-14, Accolade, Inc. v. Distinctive Software, Inc., No. 90-20202 (N.D. Cal. 1990). Note that the developer can also use the antitrust statutes to supplement the economic argument for copyright protection. For example, if a developer were required to develop new source code to perform the most basic subroutines for each software publisher, the result would be that the developer would have to deal exclusively with that particular publisher. This restriction is a form of vertical restraint and a violation of the Sherman Act as an illegal restraint of trade. A full discussion of antitrust application to software protection, however, is beyond the scope of this comment. See generally WILLIAM C. HOLMES, INTELLECTUAL PROPERTY AND ANTITRUST LAW (1991).

225. A trade secret is “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.” LEAFFER, supra note 1, at 72 (citing RESTATEMENT OF TORTS § 757 cmt. b (1939)). Copyright protection does not preempt the use of trade secret protection, but wide distribution of computer programs can make trade secret protection a less successful option. LATMAN ET AL., supra note 10, at 164.

226. LATMAN ET AL., supra note 10, at 163-64. Note that trade secret does not protect against discovery through an innocent means, such as reverse engineering. LEAFFER, supra note 1, at 72. See also Telex Corp. v. I.B.M. Corp., 510 F.2d 894 (10th Cir. 1975).

227. LATMAN ET AL., supra note 10, at 164.
secrecy can more effectively be preserved. Although trade secret has traditionally been used to protect computer programs, copyright protection is the most prudent and effective course of action for the protection of widely-distributed source code.

Occasionally patents have been used to protect software programs, but because the process is very expensive and time consuming, it does not have application to so specialized and rapidly evolving a need as the protection of source code. The examination process can take many years and cost thousands of dollars, even when an application is finally denied because it does not meet the stringent test of patentability. Furthermore, the subject matter of source code does not lend itself to patent protection, which protects "any new or useful process, machine, manufacture, or composition of matter." Patent protection has better application for a product with an anticipated long market life, as well as a significant commercial value.

Thus, because of the inherent shortcomings in other intellectual property protection, copyright is best suited to the protection of source code. With the implementation of the following proposal, copyright will provide even stronger protection for the software developer in the future, as the rights to the source code will only be transferrable upon the express grant of the developer.

V. PROPOSAL

The software developer should be able to retain the rights to source code that performs generic library subroutines. Not only is this position economically sound, but it reflects the tenets of the Constitution and the intent of the legislature as well.

228. LEAFFER, supra note 1, at 72.
229. Id.
231. LATMAN ET AL., supra note 10, at 163.
232. LEAFFER, supra note 1, at 73.
234. LEAFFER, supra note 1, at 73.
A. Copyright Protection of Source Code

The most effective way for a developer to protect the rights to the source code is to register the source code promptly with the Copyright Office at the time it is developed. In that way the source code will be protected against conflicting claims of ownership in both current and future licensing agreements. When the software developer enters into a development agreement with a software publisher, there will be no confusion as to the ownership of the underlying library routines: the developer owns the rights to the source code that performs the generic subroutines, and has prima facie proof of this ownership, and the publisher owns the source code that has been developed specifically for that contract.

1. Statutory Support for Copyright Protection

Should a conflict arise as to the ownership of the source code, the courts should look to the copyright statutes to interpret any development agreement. Specifically, section 201(d) of the 1976 Copyright Act provides for the transfer of ownership and the exclusive rights therewith and is supported by section 204(a), which stipulates that the transfer is not valid unless it is in writing and signed by the author, or authorized agent.

In addition, section 201(e) protects the author against involuntary transfer of rights, which according to the legislature “reaffirm[s] the basic principle that the United States copyright of an individual author shall be secured to that author, and cannot be taken away by any involuntary transfer.” These sections protect the developer’s rights to

236. Obviously the court should first look to the language of the development agreement to establish ownership. If the agreement expressly delivers the rights to all the source code to the software publisher, the publisher will probably prevail, barring fraud or coercion. (As previously discussed, it is very unlikely that a software developer would agree to such an arrangement, because of the economic hardship. See supra note 178 and accompanying text.) Therefore, when the language of the agreement is ambiguous or silent as to the ownership of the source code, the court should look to copyright law for assistance in interpreting the agreement.
238. Id. § 201(c).
source code library routines unless very specific requirements are satisfied by the contracting parties: Unless express language exists that explicitly transfers the rights to the entire software package to the software publisher, copyright of the underlying source code should take precedence, with the developer retaining ownership of the source code.

2. **Judicial Support for Copyright Protection**

The judiciary has repeatedly interpreted the transfer of ownership rights to mean that the rights must be expressly granted to the licensee or assignee. To date, this position has had little application in the area of software development, but is one that the courts should consistently adopt when determining ownership of source code that performs generic library routines. Thus, as in *Accolade, Inc. v. Distinctive Software, Inc.*, if the development agreement does not expressly transfer the rights to all the source code, pursuant to statute and precedent, the rights to the generic subroutines should be retained by the developer.

This interpretation of the ownership to source code is consistent with constitutional provisions and makes solid economic sense, as well. Competition is fierce in as rapidly evolving an industry as software development, and an entrepreneur needs the incentive of monetary reward before investing the time and resources necessary to develop complex source code. If the developer must contend with the potential loss of this effort every time a new development agreement is signed, there will be little incentive to stay in business, and the subsequent benefits to society will never be realized.

3. **Legislative Support for Copyright Protection**

Although the courts should find sufficient support for this position in the existing statutes, a better option is for the legislature to amend the 1976 Copyright Act, specifically section

---


201(d), concerning the transfer of ownership. Courts have interpreted this section fairly consistently to mean that the rights must be explicitly granted in the transfer agreement, but the statute only specifies that "[t]he ownership of a copyright may be transferred in whole or in part by any means of conveyance or by operation of law." 242 Certainly this vague language leaves room for judicial interpretation.

The addition of language that clearly states that any change in ownership must be expressly granted in the agreement for the transfer of rights to occur would narrow interpretation. Thus, the proposed amendment would read:

(d) TRANSFER OF OWNERSHIP. (1) The ownership of copyright may be transferred in whole or in part by the express grant of rights in any means of conveyance or by operation of law, and may be bequeathed by will or pass as personal property by the applicable laws of intestate succession. (2) Any of the exclusive rights specified by section 106,243 may be transferred as provided by clause (1) and owned separately. The owner of any particular exclusive right is entitled, to the extent of that right, to all of the protection and remedies accorded to the copyright owner by this title.

The addition of these few simple words would make it clear to all concerned that the language in an agreement must expressly grant the transfer of property rights or those rights are retained by the original owner. 244

B. Custom in the Software Industry

Finally, the courts should always consider the custom in the software industry for additional support in determining ownership rights to source code. When the developer retains the rights to the generic subroutines and the publisher retains

243. Section 106 concerns the exclusive rights in copyrighted works. Id. § 106. See supra note 16.
244. Given the history of copyright in the United States and the difficulty in securing any amendment, it is unlikely that Congress will amend the 1976 Copyright Act to incorporate so narrow a proposal. However, this proposal would resolve the confusion as to the ownership of source code in the software industry and in turn, would be an economic benefit to society as a whole. This proposal would also impact other artistic areas, such as motion pictures, screenplays, music and literature, but that discussion is beyond the scope of this comment.
the rights to the product-unique software, as is the custom, both can continue to function as viable economic entities. In turn, society as a whole benefits from technological advances through a wide range of ideas, designs, and systems, developed and produced in the most economical manner.

VI. CONCLUSION

This comment has addressed the real need for copyright protection of source code. As background for this discussion, the history of copyright law was outlined, with particular emphasis on its application to the computer and computer software. In addition, a number of landmark decisions were discussed, tracing the growth of copyright protection for computer technology. Finally, the comment analyzed the problem of shared rights to computer source code from the vantage point of the software developer and suggested a solution through copyright protection that is practical in today's expanding computer marketplace.

As computer technology continues to advance, it becomes increasingly important that the courts and legislature provide adequate protection for software innovations. Key to this effort is the software developer and the protection of the developer's rights to development tools. Currently, the software developer has the option to retain the rights to underlying library routines through copyright or trade secret protection. Although copyright offers the most consistently reliable form of protection, whichever method of intellectual property protection is selected, the developer must be able to retain the rights to this source code for reuse in subsequent development efforts.

To rule that the software publisher retains the rights to an entire software package would impact software development in the competitive market both in the United States and abroad. Additionally, it would defeat the primary purpose of the Constitution, to benefit society, as well as contradict subsequent intellectual property statutes. Clearly, it makes solid economic and legal sense for the software developer to retain the rights to the source code that performs generic subroutines: not only is the benefit passed on to the publisher and the user in the form of product variety, efficiency and price, but it serves as a powerful economic incentive to the developer to continue exploration of new technological innovations. If the United
States is to maintain a presence in the rapidly changing, critically important world of computer technology, the legal community must support this proposal and consistently interpret the laws to protect the advancement of software development.

Alexandra J. Horne