# Use and Expression: The Scope of Copyright Protection for Computer Programs, 5 Computer L.J. 433 (1985)

Leslie Wharton

## Recommended Citation

# USE AND EXPRESSION: THE SCOPE OF COPYRIGHT PROTECTION FOR COMPUTER PROGRAMS[†]

*by* LESLIE WHARTON[*]

## TABLE OF CONTENTS

Though computers were first developed in the mid-1940's, the need for copyright protection of computer programs did not arise until the 1960's. The reason is simple. The earliest computers were programmed by hand-setting electrical switches. Although the large mainframe computers of the 1950's and 1960's used software programs, the programs were custom-developed to meet the specific needs of each user. Only with the revolution in semiconductor technology in the 1970's, making small and inexpensive computers possible, did the market for computer programs expand to encompass small businesses and individual users.

---

This expansion made it impossible to produce customized programs for each user. An era of commercial software had arrived, and with it came the need for new methods of protecting the programs now sold as consumer products. Trade secret and licensing agreements had been effective for producers of software when users were closely tied to manufacturers for software and other support services, but the mass marketing of programs generally rendered both methods of protection ineffective.[1] While some software producers have sought patent protection for their programs, this approach has not been promising. In 1966, the President's Commission on the Patent System recommended that patent protection not be extended to computer programs primarily because of the anticipated administrative burden on the Patent Office.[2] Though patent protection for programs is available, few programs meet the novelty and other requirements imposed under patent law.[3] This leaves copyright as the favored system for protecting commercial computer programs.

In 1964, the Copyright Office began registering computer programs under the rule of doubt.[4] In 1974, as part of its work to amend the Copyright Act of 1909 (the 1909 Act), Congress established the National Commission on New Technological Uses of Copyrighted Works (CONTU) to make recommendations for the protection of copyrighted works when used in conjunction with computers.[5] In 1976, Congress

---

1. Trade secret protection depends on the ability of the proprietor of the secret to maintain its secrecy. *See* RESTATEMENT OF TORTS § 757 comment b (1939). When programs are sold commercially, it may be hard to show that they remain, and are intended to remain, secrets. *But see* Milgrim, *Software in Search of Protection*, 1 COMPUTER L. REP. 554, 556 (1983).

2. PRESIDENT'S COMMISSION ON THE PATENT SYSTEM, "TO PROMOTE THE PROGRESS OF . . . USEFUL ARTS" IN AN AGE OF EXPLODING TECHNOLOGY 14 (1966).

3. In Diamond v. Diehr, 450 U.S. 175 (1981), the Supreme Court established that some computer programs are entitled to patent protection. Some computer programs have been granted patent protection. *See* Paine, Webber, Jackson & Curtis, Inc. v. Merrill Lynch, Pierce, Fenner & Smith, Inc., 564 F. Supp. 1358 (D. Del. 1983). The extent of patentability for computer programs remains unclear. *See* Davidson, *Protecting Computer Software: A Comprehensive Analysis*, 23 JURIMETRICS J. 339, 348-60 (1983); Rose, *Protection of Intellectual Property Rights in Computers and Computer Programs: Recent Developments*, 9 PEPPERDINE L. REV. 547 (1982).

4. UNITED STATES COPYRIGHT OFFICE, COPYRIGHT REGISTRATION FOR COMPUTER PROGRAMS (1964), *reprinted in* 11 BULL. COPYRIGHT SOC'Y U.S.A. 361 (1964).

5. The National Commission on New Technological Uses of Copyrighted Works (CONTU) was created in 1974 to:

> study and compile data on: (1) the reproduction and use of copyrighted works of authorship: (A) in conjunction with automatic systems capable of storing, processing, retrieving, and transferring information, and (B) by various forms of machine reproduction, not including reproduction by or at the request of instructors for use in face-to-face teaching activities; and (2) the creation of new works by the application or intervention of such automatic systems or machine reproduction.

passed the Copyright Act of 1976 (the 1976 Act). While the 1976 Act made no explicit reference to computer programs, the legislative history made clear Congress' intent to include programs within the category of copyrightable "literary works."[6] In 1978, CONTU submitted its *Final Report* recommending two changes in the 1976 Act: a definition of "computer program", and a limitation on the exclusive rights of copyright owners in computer programs.[7] These recommendations were adopted by Congress in the Copyright Act of 1980 (the 1980 Act).[8] Together, the 1976 Act and the 1980 Act presumably bring computer programs within the realm of copyright subject matter. What remains uncertain is the extent of copyright protection for computer programs and whether all programs are to receive equal protection. This ambiguity in the scope of copyright protection derives largely from the nature of computer programs.

## I.  COMPUTER PROGRAMS AND COPYRIGHT: THE PROBLEM

Computer programs are developed in stages. In the first stage, the programmer usually creates a flow chart, descriptive materials, or other writings in the process of developing in algorithm (the specific steps a computer follows in performing an assigned task). For instance, if the task is to calculate the acreage and price of different parcels of real estate, the computer must obtain its data from a predetermined memory location or the keyboard, perform specific operations on those data, and display or store the results so they can be retrieved by the user. After the algorithm has been created, the programmer writes the program code, usually in a high-level programming language such as Pascal,

---

In addition, the Commission was 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." S. 3976, 93d Cong., 2d Sess. § 201, 120 CONG. REC. 30, 516 (1974), *reprinted in* NATIONAL COMM'N ON NEW TECHNOLOGICAL USES OF COPYRIGHT WORKS, 1978 FINAL REPORT 105 [hereinafter cited as CONTU, FINAL REPORT].

6.  The 1976 Copyright Act's definition of literary work is set forth at 17 U.S.C. § 101 (1982), as amended: "[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."

Congress has classified computer programs as literary works: "The term 'literary works' . . . includes catalogues, directories, and similar factual, reference, or instructional works and compilations of data. It also 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." H.R. REP. NO. 1476, 94th Cong., 2d. Sess. 54 (1976), *reprinted in* 1976 U.S. CODE CONG. & AD. NEWS 5659, 5667 [hereinafter cited as HOUSE REPORT].

7.  CONTU, FINAL REPORT, *supra* note 5, at 12.

8.  17 U.S.C. §§ 101, 117 (1982).

FORTRAN, LISP, or BASIC.[9]  Alternatively, the program may be written in Assembly language.[10]  Finally, the program is translated by yet another program into the machine code,[11] the binary (high and low voltage) signals on which the computer actually operates.[12]  These signals can be stored temporarily in Random Access Memory (RAM) or more permanently on disks, tapes, punch cards, or in Read Only Memory (ROM) chips.[13]  When written in a high-level programming language the program is called source code and can be read and interpreted by a human being with only a modicum of training.  When written in machine code it is practically unintelligible even to those trained in programming.

This transformation of a computer program from source code to machine code, from a "writing" in the literal sense to a code embodied in high and low voltage states, challenges our traditional notion of the nature of copyrightable writings.  While all other copyrighted works are created primarily to be perceived by human beings, computer programs are often entirely invisible to those who use them.[14]  In fact, many users would be appalled at the idea of purusing the thousands of lines of code which make up a spreadsheet or a word processing program.  Other literary works,[15] such as novels or poetry, are reproduced and sold for the

---

9. These high-level computer languages allow the programmer to write English-like instructions which describe the problem-solving process in steps very much like those a human would take.

10. Assembly language instructions are highly abbreviated codes such as "MOV C,A" or "JMP" which describe the problem-solving process in terms of the specific operations taken by the CPU as it moves data from memory to a temporary register, operates on the data, and "reads" the next instruction.  Assembly language is hardware specific; each microprocessor has its own set of assembly language instructions.

11. This translation from source code to machine code is performed by a compiler, interpreter, or assembler program, depending on the type of source code language used by the programmer.

12. The machine code is held in the computer in the form of high and low voltage electric currents.  The object code uses "O's" and "l's" to represent these high and low voltages.  The object code should not be confused with the machine code.  Only the machine code can be directly used by the computer; it is the machine code which instructs the CPU as to what operations to perform.

13. RAM is similar to ROM but holds information on a temporary basis only.  Disks and tapes hold information in the form of magnetized spots which are "read" by the computer as a sensitive head is passed over the magnetized areas.  A ROM chip is an array of transistor or other switches set permanently to high or low voltage states.

14. Many computer programs are sold in copy-protected forms; that is, technical means have been used to prevent the user of the program from making machine-usable copies of the program or from being able to read the program code.  Only by using a logic analyzer or microcomputer development machine can one "read" the electrical impulses of the program as they pass through the CPU and thus gain access to the program code.

15. Computer programs have been classified by Congress as literary works.  *See supra* note 6.

express purpose of being read. Sculpture is intended to be seen and even felt, painting to be seen, and music to be heard or read. No copyrighted expression, apart from computer programs, is sold to the public in a form in which the copyrighted expression is unavailable as such to the purchaser.

There is another difference between programs and other copyrighted works. Programs are created, first and foremost, to do work. They are used to perform calculations, search data files, control fuel injection systems in automobiles, and run factory assembly lines. No other form of copyrighted writing performs work in quite this way. It is true that all forms of copyrighted subject matter have some function. We may use music to alter our moods, books for education, or movies for entertainment. But in all of those uses the copyrighted work communicates something to the user, a human being. In contrast, computer programs "communicate" to computers, thereby causing the computers to perform work human beings would otherwise have to perform. Not only do programs do work, they do work in an entirely new manner. It is the program itself, the copyrighted literary work, which, when fed into a computer, performs the work.

These characteristics of computer programs are disturbing because they challenge a number of the basic axioms of copyright law and hence raise questions about the scope of copyright protection for computer programs. One such axiom is that copyright does not extend to utilitarian objects.[16] Copyright and patent, despite some overlap, are conceptually different domains, the first granting a limited monopoly to authors in their writings and the second granting quite a different set of monopoly rights to inventors in their discoveries.[17] When computer programs begin to look more like machine parts than traditional literary works, it is unclear how far the copyright in the writing should extend to the utilization of the program as an invention.

Another axiom of copyright law is that copyright protects an author's expression (that original aspect of the work created) from exploitation by others. Central to the copyright system is the notion of a copy. It is over copies of his work that the copyright owner has control, and it is his prerogative to prevent, limit, or otherwise exploit the reproduction, display, performance, distribution and sale of copies of that work. Thus, one of the basic questions in copyright law is whether one work is a copy of another. This question is especially important in the case of computer programs for two reasons. First, the transformation from human-readable source code to machine-usable code raises the question of whether the machine code is a copy of the source code pro-

---

16. *See* 17 U.S.C. § 102(b) (1982); Baker v. Selden, 101 U.S. 99 (1879).

17. *See* U.S. CONST. art. I, § 8, cl. 8.

gram. If it is not, a copyright in the source code will not protect the program in its machine code form from unauthorized reproduction. Second, we must decide whether a program embodied on a ROM chip or other medium of fixation is protected as a copy of the program. Must a copy of the program display or otherwise make available the copyrighted expression to the user, or can the program be sold in machine code versions and still be fully protected under copyright law?

Part II of this Article will explore the question of whether machine code versions of a program are copies of the source code version under current copyright law. In asking whether machine code versions are copies of the original source code, this Article considers both the relationship between source and machine codes and the form that machine code takes when held in ROM or on some other medium of fixation. After determining that machine code can be considered a copy of the source program for copyright purposes, this Article examines whether copies, to be protected under copyright law, must fulfill some communicative function—that is, whether they must allow the program user to read the copyrighted program. For policy reasons, Part II concludes that machine code versions of the program which fail to meet such a communicative function test should not be protected under copyright law.

Part III of this Article focuses on the question of whether all computer programs are equally copyrightable. Under current technology, computer programs fall into several functional categories: applications programs; operating system programs; compiler, assembler, and interpreter programs; and special-purpose machine control programs. Applications programs are designed to allow the computer user to perform a specific task, such as balancing a checkbook, playing a videogame, performing a data base search, or writing a novel. Operating system programs control the inner workings of the computer. They allow the central processing unit (CPU) to interact with peripheral devices such as the disk drive, keyboard, and cathode ray terminal, and to control the flow of data and instructions within the machine. Compiler, assembler, and interpreter programs translate the human-readable source code and data into machine code.[18] While some of these programs produce copyrightable output, and thus appear to perform some communicative role, other programs, such as operating system programs, produce no recognizable output for the computer user. They control the machine, functioning much as a machine part in that they allow the

---

18. Technically speaking, compiler, interpreter, and assembler programs are not part of the operating system; however, they have been grouped with operating system programs in recent litigation. *See* Apple Computer, Inc. v. Franklin Computer Corp., 545 F. Supp. 812, 815-16 (E.D. Pa. 1982), *rev'd*, 714 F.2d 1240 (3d Cir. 1983), *cert. dismissed*, 104 S. Ct. 690 (1984).

computer to run applications programs. Are all of these programs equally copyrightable, or should some programs be given only limited copyright protection because they are utilitarian in nature? Part III concludes that operating system and other utilitarian programs should be given only limited copyright protection so that the copyright owner cannot use his copyright to obtain a patent-like monopoly over the utilitarian function of the program.

## II. WHAT CONSTITUTES A COPY OF A COMPUTER PROGRAM?

### A. SOURCE CODE AND MACHINE CODE

The first step in determining the nature of the protected expression in a program is to establish the relationship between source code and machine code. Machine code has been characterized by CONTU and the courts as a copy of the source code.[19] Most certainly it is some sort of translation.[20] If, for copyright purposes, the machine code is a copy of the source code, then the protected element in the machine code must be that which the machine and source codes have in common. What is machine code and how does it relate to the source code from which it is derived?

Machine code is not a letter-for-letter or word-for-word encoding of the source code, but is instead a translation of instructions from human-readable programming language form to binary object code form.[21] The CPU of the computer can do little more than "read", "write", "add", "jump", "halt", and "compare."[22] Combinations of these primitive oper-

---

19. In his dissent to the majority report, Commissioner Hersey stated that the Commission had originally thought of programs in their machine-usable form as derivative works but abandoned that characterization when it realized the "invalidity of th[e] suggestion." CONTU, FINAL REPORT, *supra* note 5, at 32. The Commission then shifted to categorizing machine code versions as copies of the source code program. *Id.*.

> The Copyright Act of 1976 defines a derivative work as follows:
> [A] work based upon one or more preexisting works, such as a translation, musical arrangement, dramatization, fictionalization, motion picture version, sound recording, art reproduction, abridgment, condensation, or any other form in which a work may be recast, transformed, or adapted. A work consisting of editorial revisions, annotations, elaborations, or other modifications which, as a whole, represent an original work of authorship, is a "derivative work".

17 U.S.C. § 101 (1982). The owner of copyright has the exclusive right to prepare derivative works based on the copyrighted work. 17 U.S.C. § 106(2) (1982). It remains unclear why CONTU refused to consider machine-usable versions as derivative works based upon the original source code version of a computer program.

20. Because the source code is automatically transformed by a compiler, interpreter, or assembler program into machine code, that code is intimately and directly related to the source code version.

21. *See supra* note 12.

22. "Read" allows the CPU to receive data from memory or another source; "write"

ations allow the 8085 microprocessor, for example, to respond to seventy-eight different "assembly language" instructions.[23] Every program written in source code is ultimately and necessarily transformed into strings of these primitive instructions, encoded in binary form, on which the computer operates.[24] In this translation to machine code much of the original source code program may be lost. The programmer's commentary, which is useful for understanding all but the most simple program, is eliminated altogether.[25] In some cases other elements of the source code are not represented in the machine code version. As a result, when translated back into source code by "reverse" compiling, the original and translated source code versions, despite strong similarities, may also have noticeable differences.

What the source code and machine code have in common, then, is not the specific literary expression chosen by the programmer, but a general structure and specific strategies for bringing about an intended result. These common elements might be identified as variables, procedures, subroutines, and other steps or structures adopted in constructing a program. Some commentators have suggested that these elements be protected by copyright just as the characters, incidents, and other elements of a novel or play are protected.[26] No court has yet addressed this issue.

In some cases there is a third version of the program, the copyrightable output produces when the program is placed in the computer. Sometimes that output is almost identical with large portions of the program. This is true in computer-aided instruction and other text-producing programs where large portions of the program are text to be displayed to the user. In other cases, there is an equally strong correlation

---

allows it to send data out to memory or another peripheral device; "add" allows the microprocessor to combine two binary values in the accumulator; "jump" instructs the CPU to access an instruction other than the next one in series; "halt" brings the CPU to a halt; and "compare" allows the CPU to compare two binary values. Other operations are constructed out of these basic functions.

23. P. HOROWITZ & W. HILL, LABORATORY MANUAL FOR THE ART OF ELECTRONICS B-4 to B-6 (1981).

24. For example, the Pascal instruction "Read(x)" (where x is some character typed into the keyboard) must be translated into a series of machine code instructions that tell the computer to go to the input port (the keyboard), take the character (now encoded as a series of 0's and 1's), and place it in memory where it can later be accessed.

25. The programmer's commentary is, in essence, a set of notes written into the program which remind the programmer (and explain to any other reader) what different variables represent, the purpose of different sections of the program, and the steps being taken to solve the overall programming problem.

26. Davidson, *The Importance of Being Apple*, 2 COMPUTER L. REP. 388, 390 (1983); Davidson, Apple v. Franklin: *Facing Up to the Utilitarian Purpose Problem*, 1 COMPUTER L. REP. 754, 760-61 (1983); Pierce, *Copyright Protection for Computer Programs*, 30 COPYRIGHT L. SYMP. (ASCAP) 1, 11 (1983).

between the program code and output, except that the output is not text but rather a graphic display, musical composition, or even a spoken voice. Does the copyright in the program extend to the copyrightable output and is that output a copy of the program?

A few courts have addressed this question in the context of arcade-style videogames. In *Stern Electronics, Inc. v. Kaufman*,[27] the plaintiff had copyrighted the audiovisual aspects of his game, but not the underlying program. Responding to the defendant's contention that the only proper subject matter for copyright protection was the computer program which "determines the sights and sounds of the game's audiovisual display," the court held that a copyright in the program would not prevent a competitor from reproducing the exact audiovisual effects simply by writing a new computer program.[28] In *Midway Manufacturing Co. v. Strohon*,[29] the plaintiff had obtained copyrights for both the audiovisual output and underlying program. The court considered the computer program and audiovisual display to be distinct creations "not so 'intertwined' as to prevent their separate consideration."[30] Relying on *Stern Electronics*, the court explained that:

> [I]t is quite possible to design a game that would infringe Midway's audiovisual copyright but would use an entirely different computer program. The converse possibility . . . that a game's computer program but not its audiovisuals could be an infringement, is not foreclosed as a matter of logic and should not be as a matter of policy. The skill, ingenuity and effort that is required to design the computer program which operates the game is altogether different from the process of conceiving and designing the distinctive PAC-MAN characters.[31]

The court concluded that the videogame audiovisual display and the computer program that produced that display were separately copyrightable and that while defendant's display was not sufficiently similar to the PAC-MAN display to constitute an infringement, the computer program which produced that display did infringe Midway's copyright in the program that controlled the PAC-MAN.[32]

This solution is not altogether satisfying. While an audiovisual work may be independently conceived by an artist and then translated into program code, often it is the programmer who first creates and fixes the audiovisual display in writing the program, much as a painter creates and fixes his images with brush and paint. Separating the pro-

---

27. 669 F.2d 852 (2d Cir. 1982).

28. *Id.* at 855. Commissioner Nimmer believed that audiovisual work is not eligible for copyright protection because it is not fixed in a tangible medium of expression. CONTU, FINAL REPORT, *supra* note 5, at 27.

29. 564 F. Supp. 741 (N.D. Ill. 1983).

30. *Id.* at 749.

31. *Id.*

32. *Id.* at 742.

gram from its copyrightable output produces a "chinese box" effect, with the audiovisual display fixed in a copyrightable program which in turn is fixed in a ROM, on a disk, or on paper.[33] When the output is a literary work contained within the program or appended to it as a file, this separation of program and output is even less acceptable. Perhaps the copyrightable output should be considered a derivative work of the program, protected, at least in part, by the copyright in the program. Whether or not this solution is adopted in the future, the courts have clearly rejected any claim that the program's output is a copy of the program.

Thus the source code, the set of statements or instructions that make up the program, is the copyright-protected expression within a program. But copyright protection is not confined to the precise wording and sequence of the source code. Variables, procedures, subroutines, and other programming elements are also the author's original creation and can be protected in both the source code and machine code translation. There are several reasons for treating machine code as a copy of the original source code. First, traditional copyright treatment of literary works recognizes that the author's creation extends to the characters, incidents, and other elements of his story.[34] Second, machine code is a mechanical translation of the source code. Whatever it contains is and must be derived from the original work of authorship. Third, and most important, unless copyright protection extends to the machine-usable versions of programs, those versions can be misappropriated without any danger of infringing the copyright in the source code program.

## B. ARE ROM CHIPS COPIES OF COMPUTER PROGRAMS?

Some courts and commentators have argued that machine-usable programs, embodied on disks, tapes, or ROM chips, should be excluded from copyright protection because they are really machine parts which control the functioning of the computer. Commissioner Hersey, in his dissent to the CONTU *Final Report*, argued that:

> [Program] instructions themselves eventually become an essential part of the machinery that produces the results. They may become (in chip or hardware form) a permanent part of the actual machinery; or they may become interchangeable parts, or tools, insertable into and removable from the machine. In whatever material form, the machine-control phase of the program, when activated, enters into the computer's

---

33. According to the Second Circuit, copyright is not "defeated because the audiovisual work and the computer program are both embodied in the same components of the game. The same thing occurs when an audio tape embodies both a musical composition and a sound recording." Stern Elecs., Inc. v. Kaufman, 669 F.2d 852, 856 (2d Cir. 1982).

34. Sheldon v. Metro-Goldwyn Pictures Corp., 81 F.2d 49 (2d Cir.), *cert. denied*, 298 U.S. 669 (1936).

　　mechanical process. This is a device capable of commanding a series of impulses which open and close the electronic gates of the computer in such order as to produce the desired result.[35]

The CONTU majority agreed that computer programs serve such a utilitarian purpose but denied that this was any basis for withholding copyright protection from the program itself. The program and the machine processes are distinct. Copyright protects only the program. Anyone is free to make the computer perform the same processes.[36] Others have focused on the medium in which the machine-usable program is fixed, particularly ROM chips.[37] Here, it is argued, is the quintessential machine part, an electrical circuit made up of transistor or other switches. Since electrical circuits are not, in general, copyrightable, how can a silicon chip, which is nothing but an array of such circuits, be copyrightable?

　　The district court in *Apple Computer, Inc. v. Franklin Computer Corp.*[38] questioned the copyrightability of programs held in ROM. Judge Newcomer expressed considerable concern over whether the programmer, in writing the program, was really writing a set of instructions or merely planning the lay-out of circuitry. In the first case, he felt, the program would be a separate, copyrightable work fixed in a ROM chip. In the latter case, the program might be no more than an audiovisual work (the chip design) or a piece of engineering work.[39] Other courts have viewed the ROM chip as a mere medium of fixation which meets the statutory requirement of fixation under the 1976 Act.[40] As such, the copyright does not extend to the chip or the circuit design, just as the copyright in a book does not extend to the paper, binding, or cover.

　　ROM circuitry, under current technology, is standardized and gen-

---

35. CONTU, FINAL REPORT, *supra* note 5, at 28 (emphasis in original).

36. *Id.* at 20.

37. *See* Aufrichtig, *Copyright Protection for Computer Programs in Read Only Memory*, 11 HOFSTRA L. REV. 329 (1982); Nilles, *Copyright Protection for Programs Stored in Computer Chips: Competing with IBM and Apple*, 7 HAMLINE L. REV. 103 (1984); Stern, *Another Look at Copyright Protection of Software: Did the 1980 Act do Anything for Object Code?*, 3 COMPUTER/L.J. 1 (1981); Note, *Copyright Protection of Computer Program Object Code*, 96 HARV. L. REV. 1723 (1983).

38. 545 F. Supp. 812, 822 (E.D. Pa. 1982). The court in Data Cash Sys. v. JS & A Group, Inc., 480 F. Supp. 1063, 1066-70 (N.D. Ill. 1979) ruled that a program held in ROM was not a copy of the source code program under either the common law or the 1909 Act. This holding was compelled, the court argued, under the rule of White-Smith Publishing Co. v. Apollo Co., 209 U.S. 1 (1908). *White-Smith* was subsequently overruled by the Copyright Act of 1976. *See infra* notes 56-59 and accompanying text.

39. Apple v. Franklin, 545 F. Supp. at 822.

40. *See* Williams Elecs. Inc. v. Artic Int'l, Inc., 685 F.2d 870, 877 (3d Cir. 1982); Midway Mfg. Co. v. Artic Int'l, Inc., 547 F. Supp. 999, 1008 (N.D. Ill. 1982); Tandy Corp. v. Personal Micro Computers, Inc., 524 F. Supp. 171, 175 (N.D. Cal. 1981).

erally unpatentable.[41]  Programmers merely designate the high and low voltages they want at each addressable switch, and the chip manufacturer produces the chip as designated.[42]   There is no need for the programmer to plan the layout of the chip any more than an author needs to worry about the mechanisms of setting type or the chemical interaction of ink and paper.  Thus, copyright in a program prevents only the replication of the configuration of on and off switches which holds or represents a particular program, just as the copyright in a book prevents the replication of the precise sequence of letters, spaces, and punctuation marks that make up the text.  It does not confer a copyright on ROM chips per se and could not be used to prevent the reproduction of a ROM chip if that chip, while holding identical high and low voltage values, represented the machine-usable version of a different program.[43]

Still, some might protest that protecting the programs held in ROM is only a step away from granting copyright protection to hard-wired circuitry.  After all, programs were originally introduced into computers by a combination of hard-wiring and hand-setting the switches that made up the program.  Any program can, with sufficient time and energy, be produced by creating a sequence of different electrical circuits.  ROMs are only a convenient, manufactured form of these circuits.  Once we recognize that circuitry can hold a program—that is, a set of instructions—what is to prevent us from extending copyright protection to the electrical circuits which "instruct" room lights to turn on or off or which "instruct" an automobile engine to start?

Such questions are speculative at best.  The 1980 Act places distinct brakes on any such descent along the slippery slope into the world of utilitarian objects and patent by limiting computer programs to "statement[s] of instructions to be used directly or indirectly in a computer in order to bring about a certain result."[44]  This definition excludes electric light or ignition switches and all other forms of circuitry not used "directly or indirectly" in a computer.  It excludes the computer itself, the hardware such as the CPU, and other semiconductor circuits which

---

41. *But see* Oxman, *Intellectual Property Protection and Integrated Circuit Masks*, 20 JURIMETRICS J. 405, 432 (1980).

42. Alternatively, the programmer can use a PROM (Programmable Read Only Memory) or EPROM (Erasable Programmable Read Only Memory) in which the individual switches can be set by selectively burning out switches using pulses of high voltage.

43. Two ROM chips may hold identical arrays of switches set to identical high and low positions yet represent the machine-usable version of two different programs. This is possible because machine code is determined by the architecture of the computer and the specific operating system that controls the traffic flow from memory to CPU. Two different programs, run through different compilers, might appear to have the same machine code and yet cause the computer to perform very different tasks.

44. 17 U.S.C. § 101 (1982).

make up the computer.[45] Only programs, the sets of instructions used in the computer to bring about a certain result, can be protected under copyright law. Given the simple array of switches which make up a ROM chip, there is no reason to withhold copyright protection from the program when held in ROM or hard-wired into the computer. Extending copyright protection to the program in that form no more creates a patent-like monopoly over the circuit design than extending copyright protection to literary works in book form creates a patent-like monopoly in the art of printing and bookbinding.

## C. MUST COPIES OF A COPYRIGHTED WORK PERFORM SOME COMMUNICATIVE FUNCTION?

To qualify as a copy of a copyrighted work, it may not be sufficient that the reproduction contain the copyrighted expression of the original. Some courts and commentators claim that a reproduction must communicate the copyrighted expression to be protected as a copy of the original work; others claim that the copy must communicate something but it need not be the copyright-protected expression; and others deny that there is any communication requirement.[46] The debate is of particular importance for the computer software industry because copies of programs are frequently sold in forms that prevent the program user from accessing the program code. If copies of programs that do not communicate the program code to the user are not "copies" for copyright purposes, then others would be free to reproduce and sell those versions without infringing the author's copyright in the program.

The idea that in order to be copyrightable a work must perform some communicative function has its origin in the Copyright Clause of the Constitution which allows Congress to grant to authors "the exclusive right to their . . . writings."[47] Until the advent of computer programs, all copyrightable writings were primarily, if not entirely, communicative works. Neither Congress nor the courts has had to address the question of whether an otherwise copyrightable writing which does not communicate its protected expression to a human audience is a "writing" within the meaning of the Constitution. However, the constitutional issue may be temporarily avoided if the 1976 Act requires all copyrighted works to fulfill some communicative function.

Judicial response to the question of whether copies of computer programs must meet some communicative function test has varied con-

---

45. *But cf.* Semiconductor Chip Protection Act of 1984, Pub. L. No. 98-620, § 302, 98 Stat. 3335, 3347-56 (to be codified at 17 U.S.C. § 901) (extends copyright protection to mask works which include microprocessors and other parts of the computer hardware).

46. *See infra* notes 48-55 and accompanying text.

47. U.S. CONST. art. I, § 8, cl. 8.

siderably. Some courts have found that programs embedded in ROM chips do communicate themselves to an audience. That audience, however, is the computer. Thus, in *Tandy Corp. v. Personal Micro Computers, Inc.*,[48] the court concluded that "the imprinting of a computer program on a silicon chip, which then allows the computer to read the program and act upon its instructions" fulfilled the statutory requirement that copyrighted works be fixed in a "tangible medium of expression . . . from which they can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device."[49] In direct conflict with the *Tandy* court, the district court in *Apple Computer, Inc. v. Franklin Computer Corp.* insisted that "copyright is limited to material that can claim an underlying expressive or communicative purpose" and that the expression must be directed to a human audience.[50] However, the district court indicated that the communication requirement could be fulfilled by any expressive function and need not be a communication of the copyrighted program.[51] The Third Circuit, in reversing the lower court, denied that there was any requirement that a copy of a program embodied in ROM perform some communicative role, claiming that any such requirement was eliminated with the passage of the 1976 Act.[52] In so holding, the Third Circuit referred to its opinion in *Williams Electronics, Inc. v. Artic International, Inc.*[53] where it had argued that for copyright purposes a copy is a material object in which the copyrighted work is fixed, and that there could be no further requirement that the copy perform some communicative role.[54] Still other courts have arbitrarily claimed, somewhat metaphysically, that computer programs must always, and necessarily do, communicate with the user.[55]

These courts have based their divergent interpretations of the communication requirement on the 1976 and 1980 Copyright Acts. Is there some basis in those statutes for harmonizing the cacophony of judicial tongues?

In its opinion in *Apple Computer, Inc. v. Franklin Computer Corp.*, the Third Circuit claimed that any statutory communication requirement was eliminated when the 1976 Act overruled *White-Smith Pub-*

---

48. 524 F. Supp. 171 (N.D. Cal. 1981).

49. *Id.* at 173; *accord* Midway Mfg. Co. v. Strohon, 564 F. Supp. 741, 751 (N.D. Ill. 1983).

50. 545 F. Supp. 812, 824 (E.D. Pa. 1982), *rev'd*, 714 F.2d 1240 (3d Cir. 1983).

51. *Id.* at 825.

52. 714 F.2d at 1248.

53. 685 F.2d 870 (3d Cir. 1982).

54. *Id.* at 877.

55. Apple Computer, Inc. v. Formula Int'l, Inc., 562 F. Supp. 775, 780 (C.D. Cal. 1983).

*lishing Co. v. Apollo Co.*[56] That the 1976 Act overturned *White-Smith* is incontrovertible.[57] That it eliminated any statutory basis for requiring copies of a copyrighted work to communicate the protected expression to a human audience is far from certain.

In *White-Smith*, the Supreme Court considered whether a piano roll, the perforated sheet that, in conjunction with a player piano, produces music, infringed the plaintiff's copyright in two musical compositions. The issue was not whether the piano rolls reproduced plaintiff's music, but whether they were copies of the musical compositions. The Court held that the piano rolls were not infringing copies on the theory that:

> "A copy is that which comes so near to the original as to give to every person seeing it the idea created by the original."
>
> Various definitions have been given by the experts called in the case. The one which most commands itself to our judgment. . . defines a copy of a musical composition to be a "written or printed record of it in intelligible notation." It may be true that in a broad sense a mechanical instrument which reproduces a tune copies it; but this is a strained and artificial meaning. . . . These musical tones are not a copy which appeals to the eye. . . . A musical composition is an intellectual creation which first exists in the mind of the composer. . . . It is not susceptible of being copied until it has been put in the form which others can see and read.[58]

Under the "strained and artificial" rule of *White-Smith*, no reproduction of a copyrighted work was itself a copy unless it was directly visible to the human eye and intelligible to the human intellect. Phonograph records, videotapes, floppy disks, and ROM chips would all be excluded from copyright protection under the doctrine. However, the 1976 Act defines copyright subject matter as "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."[59] Thus, the 1976 Act abolishes the requirement that the copyrighted expression

---

56. Apple v. Franklin, 714 F.2d at 1248 (reviewing impact of Copyright Act of 1976, 17 U.S.C. § 101 (1982), on White-Smith Music Publishing Co. v. Apollo Co., 209 U.S. 1 (1908)).

57. HOUSE REPORT, *supra* note 6, at 5665, (new definition of "fixation" in section 101 is "intended to avoid the artificial and largely unjustifiable distinctions, derived from cases such as White-Smith . . . .").

58. White-Smith Music Publishing Co. v. Apollo Co., 209 U.S. 1, 17 (1908) (quoting West v. Francis, 106 Eng. Rep. 1361, 1363 (1822), *quoted with approval in* Boosey v. Whight, [1900] 1 Ch. 122, 124 (1899)).

59. 17 U.S.C. § 102(a) (1982). "Copy" is similarly defined: " 'Copies' are material objects, other than phonorecords, in which a work is fixed by any method now known or later developed, and from which the work can be perceived, reproduced, or otherwise communicated, either directly or with the aid of a machine or device." 17 U.S.C. § 101 (1982).

be visible to the eye without the intervention of a machine. It removes any requirement that it be visible as opposed to audible. However, contrary to the Third Circuit's reading, the 1976 Act does not eliminate the requirement that a "copy" of a copyrighted work must communicate.

Some courts have interpreted 17 U.S.C. §§ 101 and 102(a) as requiring only that a copyrighted program be reproduced by the computer as it operates on the program instructions.[60] Such an interpretation of the statute removes any requirement that the computer user be able to perceive the copyrighted program. However, the language "perceived, reproduced, or *otherwise* communicated" (emphasis added) in sections 101 and 102(a) belies such a reading. Rather, it suggests that the words "perceive" and "reproduce" are intended as a non-exhaustive list of those means by which the copyrighted work must be communicated. The legislative history of the 1976 Act provides an even stronger basis for maintaining that the copyrighted expression must be communicated to a human audience. In the report accompanying the 1976 Act, the House of Representatives explained that:

> [I]t makes no difference what the form, manner or medium of fixation may be—whether it is in words, numbers, notes, sounds, pictures, or any other graphic or symbolic indicia, whether embodied in a physical object or written, printed, photographic, sculptural, punched, magnetic, or any other stable form, and whether it is capable of perception directly or by means of any machine or device. . . .[61]

The House of Representatives clearly intended that the work be perceivable not to a machine, but to a *person* by means of a machine. The perceiving subject, it would appear, must be human.

Perhaps, it might be argued, the amendments made by the 1980 Act have superseded this reading of the statute. The 1980 Act added a definition of "computer program" to section 101 and a limitation on the copyright owner's exclusive rights in computer programs in section 117. Neither section directly speaks to the question of whether copies of a computer program must communicate the protected expression to their owner.[62] However, the CONTU *Final Report* recommending those

---

60. Tandy Corp. v. Personal Micro Computers, Inc., 524 F. Supp. 171, 173 (N.D. Cal. 1981); *see also* Note, *supra* note 37, at 1731.

61. HOUSE REPORT, *supra* note 6, at 5665.

62. While § 117 does not speak directly to the question of what constitutes a copyright-protected copy of a computer program, it limits the rights of the copyright owner by allowing the owner of a copy of the program to make archival copies, to create copies in the process of using the program, and to make adaptations of the program. No owner of a machine-usable copy of a program can effectively exercise the implied right to make copies and adaptations unless he has the ability to read the program code directly from the disk, ROM, or other storage medium and make alterations on the program in the computer. The CONTU *Final Report* recommends a liberal interpretation of the right to make adaptations in a program, including the right to translate it into another high-level

changes in the copyright statute also contained a discussion of the communication requirement. This is the closest thing we have to a legislative history of the 1980 Act.[63]

In recommending that copyright protection be fully extended to computer programs, CONTU relied on Judge Learned Hand's opinion in *Reiss v. National Quotation Bureau*,[64] which the Commission characterized as "the 'touchstone' for interpreting the constitutional writings requirement."[65] *Reiss* concerned the validity of a copyright in a code book containing 6,325 coined five-letter words, all of which were pronounceable, but none of which had any specific meaning. Purchasers of the code book could assign meanings to the words and use them for private long-distance cable transactions. Judge Hand asked whether these coined words qualified as writings under the Constitution. For Hand, the critical issue was whether a writing had to have a predetermined meaning. If so, the code book could not be copyrighted.[66] But, Hand argued:

> Not all words communicate ideas; some are mere spontaneous ejaculations. Some are used for their sound alone, like nursery jingles. . . . Conceivably there may arise a poet who strings together words without rational sequence . . . through whose beauty, cadence, meter, and rhyme he may seek to make poetry. . . .
>
> Works of plastic art need not be pictorial. They may be merely patterns, or designs, and yet they are within the statute.[67]

He concluded, "If . . . models or paintings are 'writings,' I can see no reason why words should not be such because they communicate nothing."[68] Judge Hand might better have concluded that there was no reason why they could not be copyrighted because they communicated *nothing in particular*. *Reiss* stands not for the proposition that copyrightable writings need not be communicative or expressive in function, but for the much more narrow proposition that to qualify as a writing the work need not have a designated meaning which it communicates.

---

programming language, the right to add features to the program, and the right to make other transformations necessary to make the program fulfill the user's needs. CONTU FINAL REPORT, *supra* note 5, at 13.

63. The CONTU *Final Report, supra* note 5, has been treated by courts as a statement of the legislative intent behind the 1980 Act. *See* Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1252 (3d Cir. 1983); Apple Computer, Inc. v. Formula Int'l, Inc., 562 F. Supp. 775, 781 (C.D. Cal. 1983). Some commentators have questioned the appropriateness of adopting the CONTU *Final Report* as legislative history. *See* Elman & Moskowitz, *Comment on the Third Circuit's Decision in* Apple v. Franklin, 2 COMPUTER L. REP. 391, 392 (1983).

64. 276 F. 717 (S.D.N.Y. 1921).

65. CONTU, FINAL REPORT, *supra* note 5, at 14.

66. 276 F. at 718.

67. *Id.*

68. *Id.* at 719.

Just as it is absurd to ask for the meaning of a Joan Miro painting or a jazz improvisation, so we cannot require, as a constitutional matter, that literary works be representational or otherwise "meaningful" to qualify for copyright protection.[69]

While CONTU's ultimate position remains uncertain, it never maintained that a copy of a computer program need not communicate its copyrighted expression to a human audience. Rather, the CONTU *Final Report* indicates that programs (fixed in whatever form) must perform some communicative function to qualify for copyright protection.[70] But CONTU refused to limit copyright protection to those programs which produced copyrightable output. To exclude programs which performed useful functions "would be inconsistent 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."[71] Nevertheless, CONTU's description of the ways in which a copyrighted work could be infringed suggests its intention that to be copyrightable a work must be capable of being perceived by the program user:

> A computer program may be misappropriated in a variety of ways. In the first and most straightforward instance, the program listing or the programmer's original coding sheets might be photocopied, which would clearly be an infringement. The unarguably copyrightable writing has been taken. But, what if the program, rather than being recorded on paper, is recorded on magnetic tape or disk? If the tape is used without authorization to produce a printed, human-readable version of the program, again an infringement has occurred. Should the result be different if the tape is copied? That copy may still be used to prepare a printed version at will. There is a one-to-one correspondence between the printed characters on paper and the magnetized areas of the tape. The tape is simply a version of the program from which a human-readable copy may be produced with the aid of a machine or device.[72]

---

69. This would support the argument that machine code, though unintelligible, is copyrightable.

70. In distinguishing programs from mere machine parts, the Commission compared programs with videotapes and phonograph records and announced that:

> All three types of work are *capable* of communicating with humans to a far greater extent than the coined words discussed by Judge Hand in *Reiss v. National Quotation Bureau*. In all three instances, the medium in which copyrighted material is stored is moved past a sensing device at a set speed, causing electric current to flow, and ultimately resulting in the movement of machine parts to print words, display pictures, or create sounds.

CONTU, FINAL REPORT, *supra* note 5, at 21 (emphasis in original).

71. *Id.*

72. *Id.* at 22.

When is a version of a program not a copy for copyright purposes? According to CONTU:

> Only when the program is inserted—instruction for instruction—into the processing element of the computer and electrical impulses are sent through the circuitry of the processor to initiate work is the ability to copy lost. . . . If it should be possible to tap off these impulses then, perhaps, the process would be all that was appropriated, and no infringement of the copyright would occur.[73]

The ability to make a human-readable version is critical in CONTU's view. Only when that ability is lost, presumably at the stage where the program actually controls the computer's functioning, would a version of the program not qualify as a copyright-protected copy.[74]

Is a version of a copyrighted program considered a copy, and therefore protected against infringement, when it is merely capable of producing human-readable version, or must it do so in the ordinary course of use? Neither CONTU nor the definitions of "copy" and copyright subject matter provided in the 1976 Act answer this question. The answer, if any, must be found in the policies and values behind our copyright law.

Since it is technically possible to "break into" any medium of fixation and retrieve the program code, it could be that any instantiation of a computer program could itself qualify as a protected copy. This interpretation of the communication requirement comports well with a policy of protecting computer programs as items which derive their value from use and not as literary works to be read or viewed by the public. It is also consistent with current copyright law which protects all writings from the moment of first fixation. A manuscript, hidden away in a desk drawer, is as much protected by copyright as a best seller available at every newsstand. There are practical reasons for extending copyright protection to copy-protected versions of a program. Many fear that revealing the underlying programs to computer users will only promote infringement in a technology where it is far easier and less expensive to make illicit copies than to purchase legitimate ones in the market. Enforcement of the copyright laws against such infringers is next to impossible. But extending copyright protection to copies of a program from which the copyrighted expression cannot be perceived in the normal course of its use is tantamount to using copyright law to protect what remains essentially a trade secret. Though the unpublished manu-

---

73. *Id.*

74. While CONTU's analysis of the electronic processes taking place in the computer leaves something to be desired—the program embedded in a disk or on ROM can only be copied by transforming the program into a series of electrical signals and, in a sense, tapping them off—the example was intended to distinguish the copyrightable program from the use of the program which cannot be enjoined under the rule of *Baker v. Selden.*

script is nominally protected under copyright law, its owner reaps no benefit from that protection until she makes it public.[75] The computer software vendor who uses her copyright to obtain a monopoly on the sale of a program while preventing purchasers from gaining access to the program code itself, is using copyright law to ensure financial gain where trade secret protection alone would fail.

The limited monopoly granted to authors under copyright law has a quid pro quo—the public benefit. Unitl now, that public benefit clearly has been the creation and dissemination of new creative expression and the ideas and useful learning it embodies. With computer programs there is a sharp bifurcation between the benefits the public reaps from being able to use the work and from being able to read the program. While we are accustomed to thinking of computer programs as useful articles, there is also enormous public benefit to be gained from publishing programs and making the knowledge and techniques contained therein available to other programmers and program users. Program writing, after all, is itself a useful art that may be promoted by publishing examples for study as well as use. Computer programming is no longer the exclusive territory of a specialized group called computer scientists. Children, college students, executives, and others are writing programs in more than a dozen computer languages and dialects. This enormous growth in what is called "computer literacy" suggests that computer programs are more than mere tools which can be employed without the need to understand them. They are a new form of expression, of conceptualizing and solving practical problems, and of communicating ideas.[76] Copyright used to enforce trade secret protection for computer programs would stifle this development at tremendous social cost. Certainly, extending copyright protection to copies of programs which are copy-protected would undermine the policy behind section 117 of the 1980 Act which explicitly allows owners of copies of a computer program to adapt and copy the program. CONTU, in recommending adoption of section 117 by Congress, broadly interpreted the rights of the owner of a copy of a computer program.[77] In order to ex-

---

75. The 1976 Act states that copyright "subsists" in works of authorship from the moment of fixation in a tangible medium of expression, thereby eliminating the publication requirement that was a prerequisite for federal copyright protection under the 1909 Act. One reason for this change was to avoid the frequent and vexatious litigation over when publication first occurred. It is arguable that the copyright statute no longer makes public dissemination a quid pro quo for copyright protection; however, the financial rewards made available to the author by the copyright system can only be realized if the author does publish the work.

76. COMPUTER LITERACY: ISSUES AND DIRECTIONS FOR 1985 (R. Anderson, B. Hunter, R. Seidel eds. 1982).

77. Because of the lack of complete standardization among programming languages and hardware in the computer industry, one who rightfully acquires a

ercise those rights, the owner of a copy of the program must have direct access to the program code.

In that delicate balancing of public and private benefits which underlies all of copyright law, the scales ought to be tipped in favor of the public.[78] At issue here is not the creation and dissemination of computer programs per se, but the form in which they are published. If programs are sold in copies from which the program code can be obtained only at great expense or effort, the public effectively loses access to that copyrighted expression for all purposes other than use in conjunction with a computer. Even that use is limited because users cannot modify the program to serve their individual needs. What the copyright owner gains from the sale of copy-protected versions of the program is an extra edge in preventing copyright infringement. That copy-protection will not deter commercial competitors from discovering and even replicating the program. At most it will prove useful against individuals who make extra copies of a program for non-commercial uses.

The policy question which must be addressed by Congress and the courts is whether copyright law will protect computer programs simply because they are useful articles, using the copyrighted literary expression solely to identify the protected work, or whether it will limit copyright protection to works which not only contain but also make public the protected expression. Some guidelines can be obtained from the treatment of pictorial, graphic, and sculptural works in the 1976 Act. Useful articles are not copyrightable.[79] A useful article is "an article having an intrinsic utilitarian function that is not merely to portray the appearance of the article or to convey information."[80] Under the 1976 Act, the design of a useful article will be protected by copyright "if, and only to the extent that, such design incorporates pictorial, graphic, or sculptural features that can be identified separately from, and are capa-

---

copy of a program frequently cannot use it without adapting it to that limited extent which will allow its use in the possessor's computer. The copyright law, which grants to copyright proprietors the exclusive right to prepare translations, transformations, and adaptations of their work, should no more prevent such use than it should prevent rightful possessors from loading programs into their computers. Thus, a right to make those changes necessary to enable the use for which it was both sold and purchased should be provided. The conversion of the program from one higher-level language to another to facilitate use would fall within this right, as would the right to add features to the program that were not present at the time of rightful acquisition. These rights would necessarily be more private in nature than the right to load a program by copying it and could only be exercised so long as they did not harm the interests of the copyright proprietor.

CONTU, FINAL REPORT, *supra* note 5, at 13 (footnote omitted).

78. Twentieth Century Music Corp. v. Aiken, 422 U.S. 151 (1975) (copyright monopoly must ultimately serve the public good by promoting broad public availability).

79. 17 U.S.C. § 102(b) (1982).

80. 17 U.S.C. § 101 (1982) (definition of "useful article").

ble of existing independently of, the utilitarian aspects of the article."[81]

Computer programs, when placed in a computer, are useful articles. They perform work. The program itself, however, is literary in nature. As a literary work it is subject to copyright protection. The statutory treatment of pictorial, graphic, and sculptural works suggests that computer programs should be protected only insofar as they are separable from their utilitarian use. Copies of programs that perform useful work but which do not independently display or otherwise allow access to the program itself would be treated as useful articles outside the scope of copyright protection.

Machine-usable versions of a computer program are copyright-protected copies of that program to the extent that the source code is reproduced in the machine code and the program can be "reproduced, perceived, or otherwise communicated" to the human user of the program.[82] Policy considerations suggest that machine code versions that prevent access to the program code in the normal course of use be denied protection under copyright law. But within these strictures, it appears that machine code programs are fully copyrightable and that a copyright in the original source code will protect the machine code program as well.

## III. COMPUTER PROGRAMS AS UTILITARIAN OBJECTS: RETHINKING *BAKER v. SELDEN*

### A. THE UTILITARIAN OBJECT PROBLEM

The 1980 Act defines a computer program as a "set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result.[83] Some programs function to produce copyrightable output, others control computer operations, and yet others are used to run machinery from robots to microwave ovens. Should programs be distinguished according to these functions and granted differential copyright protection on that basis?

A computer standing alone can do nothing. Computer programs transform that general purpose machine into a specific machine, one that can perform spreadsheet analyses, create a painting, or act as a control device for an industrial robot.[84] In this sense, all programs are

---

81. 17 U.S.C. § 101 (1982) (definition of "pictorial, graphic, and sculptural works"). Copyright protection extends to pictorial, graphic, or sculptural works which are "reproduce[d] in or on any kind of article, whether useful or otherwise." 17 U.S.C. § 113(a) (1982).

82. 17 U.S.C. §§ 101, 102(a) (1982); *see supra* text accompanying notes 60-61.

83. 17 U.S.C. § 101 (1982).

84. Oxman, *supra* note 41, at 431 ("The ROM is what makes the computer in question unique. The ROM . . . amounts to a permanent on-board program which transforms a

blueprints for machines. But the program is not only a blueprint. It performs the dual function of blueprint for the special purpose machine it creates and the means by which that machine operates. Viewed from this perspective, computer programs appear much more like machines than copyrightable writings.

Many are unconcerned by the machine-like nature of computer programs. The CONTU majority based its strong support of copyright protection for computer programs on the grounds that:

> [Programs] are used in an almost limitless number of ways to release human beings from such diverse mundane tasks as preparing payrolls, monitoring aircraft instruments, taking data readings, making calculations for research, setting type, operating assembly lines, and taking inventory. . . . For both economic and humanitarian reasons, it is undesireable for people to carry out manually the process described in painstaking detail in a computer program. Machines, lacking human attributes, cannot object to carrying out repetitious, boring, and tedious tasks. Because machines can and do perform these tasks, people are free to do those other things which they alone can do or in which they find a more rewarding expenditure of their efforts.[85]

Some courts argue that Congress, in adopting the amendments recommended by CONTU, also adopted CONTU's view that copyright protection should extend to all computer programs regardless of their function.[86] The position that utilitarian use does not preclude copyright protection also finds support in the Supreme Court opinion in *Mazer v. Stein*[87] which stated, in dicta, that "nothing in the copyright statute . . . [supports] the argument that the intended use or use in industry of an article eligible for copyright bars or invalidates its registration."[88] Since programs are copyrightable as literary works, their use in a computer,

---

potentially general-purpose machine (the microcomputer with an unprogrammed ROM) into a special purpose machine."); *see also* Gemignani, *Should Algorithms Be Patentable?*, 23 JURIMETRICS J. 326, 331 (1982).

85. CONTU, FINAL REPORT, *supra* note 5, at 10.

86. Apple Computer, Inc. v. Formula Int'l, Inc., 725 F.2d 521, 524-25 (9th Cir. 1984) ("As recommended by the CONTU majority, the [1980] 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."); Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1252 (3d Cir. 1983) ("[W]e can consider the CONTU Report as accepted by Congress since Congress wrote into the law the majority's recommendation almost verbatim. . . . the statutory definition of a computer program . . . makes no distinction between application programs and operating programs") (citations omitted); Apple Computer, Inc. v. Formula Int'l, Inc., 562 F. Supp. 775, 781 (C.D. Cal. 1983) ("It is crystal-clear that CONTU recommended that *all* computer programs . . . be included within copyright protection. There likewise can be no doubt but that Congress accepted that recommendation and embodied it in the 1980 amendments to the Copyright law.") (emphasis in original).

87. 347 U.S. 201 (1954).

88. *Id.* at 218.

whatever the intended result, should not divest them of copyright protection.

Yet there are others who remain uncomfortable with this position. Commissioner Nimmer, in his concurring opinion in the CONTU *Final Report*, suggested that computer programs be distinguished according to their results; programs that control automobile fuel injection systems, temperature levels in buildings, or traffic signals would not be eligible for copyright protection.[89] Nimmer believed such a distinction necessary to prevent the Copyright Act from becoming "a general misappropriation law, applicable as well in what has traditionally been regarded as the patent arena, and, indeed, also in other areas to which neither copyright nor patent law has previously extended."[90] A similar functional distinction has been suggested in recent litigation over the copyrightability of operating system programs, the programs which control the computer's internal operations.[91]

Computer programs can be divided into three basic types according to their function. "Type I" programs (applications programs) allow the user to perform "intellectual" work. These programs usually produce copyrightable output in the form of written text, graphic displays, or even musical compositions. "Type 2" programs (operating system programs) produce no recognizable output. They run the computer, allowing it to receive input from a keyboard or other peripheral device, display output, translate higher level programming languages into machine-usable code, and perform other similar tasks.[92] Without such programs, the computer would be a useless piece of hardware. "Type 3" programs, like type 1 programs, produce identifiable output, but the output is "physical" work instead of "intellectual" work. Type 3 programs may be used in general purpose computers or special purpose devices to control microwave ovens, traffic lights, and to control feedback systems in chemical and industrial processes.

No court has yet addressed the question of whether type 3 programs are fully copyrightable. Only a few courts have considered the

89. CONTU, FINAL REPORT, *supra* note 5, at 27.

90. *Id.* at 26.

91. In both *Apple v. Franklin* and *Apple v. Formula*, the defendants argued that the operating system programs which they allegedly had copied were not copyrightable under the 1976 and 1980 Acts. They based their claim on the fact that, unlike applications programs, operating system programs control the internal operations of the computer and therefore are utilitarian processes or methods of operation and that they do not communicate directly with the computer user. Apple Computer, Inc. v. Formula Int'l Corp., 725 F.2d 521 (9th Cir. 1984); Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983). *See also*, Brief for Appellee at 25, Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983), *reprinted in* 1 COMPUTER L. REP. 681, 691-92 (1983).

92. *See supra* note 18.

copyrightability of type 2, operating system, programs. One recent case, *Apple Computer, Inc. v. Franklin Computer Corp.*[93] merits close examination both because of the effect that case will have on future litigation and because the arguments raised on both sides must be taken into account in any determination of the issue.

## B.  OPERATING SYSTEM PROGRAMS:  *APPLE V. FRANKLIN*

In *Apple Computer, Inc., v. Franklin Computer Corp.*[94] the heart of the defendant's argument was that Apple's copyright on its operating system programs gave it a monopoly on the machine that runs Apple-compatible software. Franklin claimed that the operating system programs constituted a process, procedure, or method of operation excluded from copyright protection under section 102(b) of the 1976 Act.[95] By obtaining a copyright on these programs, Apple effectively was able to prevent anyone else from creating a computer capable of running the numerous applications programs developed by third party programmers to run on the Apple machine. This monopolization of a machine (and its software) was possible, Franklin alleged, because there were only a limited number of ways to write an operating system that could run Apple-compatible software.[96] This merger of "idea" and "expression", of operating system and program code, made it possible for Apple to use its copyright in the program codes to create an impermissible monopoly in a computer system.

Apple's position, in defense of its copyright, was that there is no adequate way to distinguish applications programs (type 1) from operating system programs (type 2) since both make the computer perform

---

93.. 545 F. Supp. 812 (E.D. Pa. 1982), *rev'd,* 714 F.2d 1240 (3d Cir. 1983), *cert. dismissed,* 104 S.Ct. 690 (1984).

94. *Id.*

95. The relevant statutory language is reprinted in the text accompanying note 100, *infra.*

96. The factual dispute over how many ways a program can be written to achieve the same result has not been fully addressed by any court. CONTU received testimony that there are, in principle, a near infinite number of ways to write any program. CONTU, FINAL REPORT, *supra* note 5, at 20 n.106. In *Apple v. Franklin,* Apple did not try to argue that its operating system programs could be rewritten to achieve Apple-software compatibility, but merely argued that many different operating system programs could be written for the Apple computer. *See* Elman & Moskowitz, *A Personal View of the* Apple v. Franklin *Appellate Arguments—A Hundred Minutes on Software Copyright,* 1 COMPUTER L. REP. 901, 904 (1983). In *Apple v. Formula,* Apple's witness admitted that the operating system programs could be rewritten to achieve 98% compatibility with Apple computers. Apple Computer, Inc. v. Formula Int'l Corp., 562 F. Supp. 775, 782 (C.D. Cal. 1983), *aff'd,* 725 F.2d 521 (9th Cir. 1984). But, in *Apple v. Franklin,* Franklin's witness said that no rewritten operating system could achieve 100% compatibility. Brief of Appellee at 20, Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983), *reprinted in* 1 COMPUTER L. REP. 681, 695 (1983). *See infra* note 128.

work.[97] Since both are utilitarian and both work together to bring about a given result for the computer user, both may be classified as processes or methods of operation. To attempt to draw a line between the two types of programs would be to question the copyrightability of all computer programs. According to Apple, it would be better to allow copyright protection for all programs than to risk undermining Congress' clear intent to provide copyright protection for at least some programs.

The Third Circuit, in upholding the copyrightability of Apple's operating system programs, agreed with Apple that applications and operating system programs are indistinguishable and denied that either type of program constitutes a process, system, or method of operation excluded from protection under section 102(b) of the Copyright Act. Citing the CONTU majority report, the court argued:

> Since it is only the instructions which are protected, a "process" is no more involved because the instructions in an operating system program may be used to activate the operation of the computer than it would be if instructions were written in ordinary English in a manual which described the necessary steps to activate an intricate complicated machine. There is, therefore, no reason to afford any less copyright protection to the instructions in an operating system program than to the instructions in an application program.[98]

As to Franklin's argument, the court denied that there was any merger of idea (that of running Apple-compatible software) and expression (the operating system programs) by refusing to accept Franklin's position that running Apple-compatible software was the relevant "idea":

> The idea which may merge with the expression, thus making the copyright unavailable, is the idea which is the subject of the expression. The idea of one of the operating system programs is, for example, how to translate source code into object code. If other methods of expressing that idea are not foreclosed as a practical matter, then there is no merger. Franklin may wish to achieve total compatibility with independently developed application programs written for the Apple II, but that is a commercial and competitive objective which does not enter into the somewhat metaphysical issue of whether particular ideas and expressions have merged.[99]

How do the positions taken by the parties and court in *Apple v. Franklin* comport with copyright law under the 1976 and 1980 Acts? To answer this question it is necessary to look at section 102(b) of the 1976

---

97. Appellant's Opening Brief at 30-33, Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983) *reprinted in* 1 COMPUTER L. REP. 655, 674-75 (1983).

98. Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1251 (3d Cir. 1984).

99. *Id.* at 1253.

Act and the origins of the rule that copyright protection does not extend to utilitarian objects.

### C.  THE RULE OF *BAKER V. SELDEN*

Section 102(b) of the 1976 Act completes the statutory definition of copyright subject matter with an exclusionary clause: "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."[100]  This section is unbending in its denial of copyright protection for any work or any part thereof which is an idea, procedure, or method of operation *regardless of whether it would otherwise qualify as a copyrightable writing*.  In explaining the applicability of this provision to computer programs, the House of Representatives reported that:

> Copyright does not preclude others from using the ideas or information revealed by the author's work. It pertains to the literary[,] musical, graphic, or artistic form in which the author expressed intellectual concepts. . . .
>
> Some concern has been expressed lest copyright in computer programs should extend protection to the methodology or processes adopted by the programmer, rather than merely to the "writing" expressing his ideas. Section 102(b) is intended, among other things, 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 the copyright law.
>
> Section 102(b) in no way enlarges or contracts the scope of copyright protection under the present law. Its purpose is to restate, in the context of the new single Federal system of copyright, that the basic dichotomy between expression and idea remains unchanged.[101]

What then are the ideas, methods, processes, or systems excluded from copyright protection under section 102(b), and how are we to distinguish the programmer's expression from the idea that is expressed?

The fundamental axiom that copyright does not cover processes, methods, systems, or other utilitarian works has its root in the Supreme Court decision in *Baker v. Selden*.[102]  In that case, the Court considered respondent's book explaining a new accounting system with examples showing the arrangement of columns and headings necessary to employ the system.  The Court held that Selden could copyright the description or explanation of his accounting system but not the "ruled lines and

---

100.  17 U.S.C. § 102(b) (1982).
101.  HOUSE REPORT, *supra* note 6, at 5670.
102.  101 U.S. 99 (1879).

headings" of the T-accounts since they were necessary for the use of the system.[103] "Though *Baker v. Selden* stands for several fundamental principles,"[104] it is generally conceded that, at minimum, the case stands for the proposition that a copyright on a description or explanation of a system, process, method, or other useful art cannot confer a monopoly on the use of the system itself.[105] Thus, the Court in *Baker* explained that a book on the "composition and use of medicines" or on "the construction and use of ploughs, or watches, or churns" cannot confer an exclusive right in the author to make medicines, ploughs, watches, or churns according to the explanation in the book.[106] The policy behind this distinction goes to the heart of the copyright clause of the Constitution: "The very object of publishing a book on science or the useful arts is to communicate to the world the useful knowledge which it contains. But this object would be frustrated if the knowledge could not be used without incurring the guilt of piracy of the book."[107] Such exclusive rights to employ the art would effectively allow patent rights to be obtained without the rigorous registration and other requirements of the patent system.

The distinction between use and explanation seems clear when applied to the physical arts and sciences like medicine or plough-making but it becomes murky in cases like *Baker* itself where the practice of the art described entails the reproduction of the exact words, illustrations, or charts contained in the copyrighted book. In *Baker*, the Supreme Court held that copyright would not extend to those parts of the copyrighted work *necessary* for the practice of the art, declaring:

> And where the art it teaches cannot be used without employing the methods and diagrams used to illustrate the book, or such as are similar to them, such methods and diagrams are to be considered as necessary incidents to the art, and given therewith to the public; not given

---

103. *Id.* at 105. ("The description of the art in a 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.")

104. Franklin argued before the Third Circuit that *Baker v. Selden* stands for three distinct propositions:

> *First, Baker* teaches that use of a system itself does not infringe a copyright on the description of the system. *Second, Baker* enunciates the rule that copyright does not extend to purely utilitarian works. *Finally, Baker* emphasizes that the copyright laws may not be used to obtain and hold a monopoly over an idea. *Baker* highlights the principal difference between the copyright and patent laws—a difference that is highly pertinent in this case.

Brief for Appellee at 22, Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir.), *reprinted in* 1 COMPUTER L. REP. 681, 696 (1983).

105. M. NIMMER, NIMMER ON COPYRIGHT § 2.18(B) (1978).

106. Baker v. Selden, 101 U.S. at 102.

107. *Id.* at 103.

> for the purpose of publication in other works explanatory of the art,
> but for the purpose of practical application.[108]

On this basis, the Court held Selden's T-account forms uncopyrightable.

Closely aligned with the rule of *Baker v. Selden* is the rule that copyright protects only the author's expression and not the idea which the author expresses. Thus, "[w]hen the 'idea' and its 'expression' are . . . inseparable, copying the 'expression' will not be barred, since protecting the 'expression' in such circumstances would confer a monopoly of the 'idea' upon the copyright owner free of the conditions and limitations imposed by patent law."[109] The rule that copyright in an explanation, description, or other expression of a useful system will not extend to the use of that system and the rule that where an idea can be expressed in only a limited number of ways no copyright will be allowed to create a monopoly in that idea are conceptually different and require different inquiries.[110] Where computer programs are at issue, these rules must be applied in a two-step process. The first step is to determine, under *Baker*, whether the program is utilitarian and, if so, what its particular function is (that is, what system, method, or process it performs). The second step is to see how many programs could be written to express that same system, method, or process without infringing the copyright in the first program.

In *Apple Computer, Inc. v. Franklin Computer Corp.*,[111] the Third Circuit held that an operating system should be defined in broadly functional terms, as a system to "translate source code into object code" and not, for instance, to translate a specific source code language into the machine code appropriate for a given computer architecture.[112] Since the operating system was defined in only general terms, the Third Circuit found no merger between the program's expression (the sequence of instructions) and the "idea" or function of the program. The court concluded that since many different programs could be written to process properly configured applications programs, it was not necessary for Franklin to copy or closely replicate Apple's operating system programs in order to create its own system.[113]

In deciding that Apple's operating system programs could be

---

108. *Id.*

109. Herbert Rosenthal Jewelry Corp. v. Kalpakian, 446 F.2d 738, 742 (9th Cir. 1971).

110. *Baker* has been interpreted as granting copyright protection "only for the expression of the idea—not the idea itself." Mazer v. Stein, 347 U.S. 201, 217 (1954). The *Baker* court, however, never mentioned an idea-expression dichotomy; instead, it distinguished between description or explanation and use. Nevertheless, the word "idea" has become a generic term encompassing all the exceptions to copyright protection contained in § 102(b) of the 1976 Act.

111. 714 F.2d 1240 (3d Cir. 1983).

112. *Id.* at 1253.

113. *Id.*

treated as generic systems, the Third Circuit relied on the test for distinguishing between copyright-protected artistic expression and uncopyrightable ideas developed by Judge Learned Hand in *Nichols v. Universal Pictures Corp.*[114]

> 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.[115]

Thus, the Third Circuit applied a test intended for deciding the scope of an author's expression in a literary work to determine the process, method, or system of which Apple's operating system program was an expression. In applying this test, the Court failed to recognize that the idea-expression test is appropriate only as a second-level inquiry where utilitarian works are concerned. The system or method which the program "expresses" must first be determined, as an examination of the *Baker* decision and its progeny makes clear.

In *Baker*, the plaintiff had developed a "peculiar" bookkeeping system in that it allowed one to present the "entire operation of a day, a week, or a month, on a single page, or on two pages facing each other."[116] The system was by no means the only available bookkeeping system, nor did it monopolize the art of bookkeeping. It was, to say the least, one among many alternative bookkeeping systems, a fact demonstrated by defendant Baker's ability to achieve the same results using different account forms. The Supreme Court denied Selden a copyright in the account forms not on the ground that such a copyright would prevent either bookkeeping in general or the best method of bookkeeping, but instead denied the copyright because granting it would prevent the use of Selden's own bookkeeping system, his invention, which he had dedicated to the public by publishing it without first obtaining a patent on the system.[117] The Supreme Court never subjected Selden's accounting system to scrutiny to see if it monopolized some more ge-

---

114. *Id., citing* Nichols v. Universal Pictures, 45 F.2d 119 (2d Cir. 1930).

115. Nichols v. Universal Pictures, 45 F.2d at 121 (citation omitted).

116. Baker v. Selden, 101 U.S. 99, 100 (1879).

117. *Id.* at 104. Some, relying on subsequent cases such as Brown Instrument Co. v. Warner, 161 F.2d 910 (D.C. Cir. 1947) and Taylor Instrument Cos. v. Fawley-Frost Co., 139 F.2d 98 (7th Cir. 1943), have read *Baker* to deny a copyright for Selden's T-account forms because they lacked copyrightable expression. But the *Baker* court never gave such a basis for its decision. Judge Newcomer in his opinion in *Apple v. Franklin*, invoked what he called the "Baker-Taylor" doctrine, claiming that copyright lies only in that which teaches or explains and not that which is intended for use. 545 F. Supp. at 821.

neric bookkeeping function. The Court took Selden's system as such and only then asked whether there was an impermissible merger of the system (use) with the expression (account forms) adopted by the author.

A glance at other cases dealing with utilitarian writings in which, unlike the *Baker* case, the use of the system was inseparable from the copyrighted expression,[118] confirms that the system or method expressed in a writing is not subject to further abstraction by the court. In *Crume v. Pacific Mutual Life Insurance Co.*,[119] the court considered the alleged infringement of four pamphlets describing a "plan or method by which insolvent life insurance companies may be reorganized through mutualization and readjustment of liabilities."[120] The court undertook no inquiry to determine whether there were other methods by which life insurance companies could be saved from bankruptcy or whether there were other possible plans for "mutualization and readjustment of liabilities." Rather, the court proceeded directly to the question of whether the words contained in plaintiff's pamphlets were necessary for the use of the specific plan contained in these pamphlets. Though the court found the defendant's work had sufficient dissimilarity to avoid the charge of infringement, it set out a rule limiting any copyright in any language that set forth such a plan where the plan could be effected solely by the use of the words descriptive thereof.[121]

---

118. In *Baker*, the T-account forms were entirely separable from the text describing how to use the new accounting system. The Second Circuit has distinguished *Baker* in cases where the copyrighted work contains copyrightable expression inextricably intertwined with the "writings" necessary for the use of the system described. *See* Continental Casualty Co. v. Beardsley, 253 F.2d 702, 704 (2d Cir. 1958); Harcourt, Brace & World, Inc. v. Graphic Controls Corp., 329 F. Supp. 517, 524 (S.D.N.Y. 1971). Other courts have not made this distinction, a distinction which appears to have little practical effect on the outcome of the case.

119. 140 F.2d 182 (7th Cir. 1944).

120. *Id.* at 182.

121. [P]laintiff recognizes defendant's right to the use of the plan or method taught by plaintiff, but denies to the defendant the right to use the words necessary to effect such use. It appears to us that the concession is inconsistent with the denial. Plaintiff attempts to compare the instant situation with books containing plans and descriptions for houses, or plans for formal gardens, or plans for interior decoration. In such instances, however, the disclosure of the plan may be put to use by the utilization of material, tools, and equipment in the hands of the mechanic or workman. The use to which the public is entitled is effected by means other than the embodiment of words. In the instant situation there is no room for the skill of the mechanic or artisan in utilizing the plan or the method disclosed. Its use, to which the public is entitled, can be effected solely by the employment of words descriptive thereof. In our view, where the use can be effected only in such manner, there can be no infringement even though the plan or method be copied. We realize that such a view leaves little, if any, protection to the copyright owner; in fact, it comes near to invalidating the copyright. This situation, however, results from the fact that the practical use of the art explained by the copyright and lodged in the public domain can be at-

Similarly, in *Morrissey v. Proctor & Gamble Co.*,[122] the court recognized that a sweepstakes based on social security numbers was an uncopyrightable system. It did not engage in an inquiry as to alternative types of sweepstakes, based, for instance, on telephone numbers, but asked only if there was a merger between the idea (the specific sweepstakes system) and the expression contained in plaintiff's copyrighted rules for the sweepstakes contest.[123]

Operating system programs, such as the plans in *Crume* and the sweepstakes rules in *Morrissey*, are classified under copyright law as literary works.[124] The programs, like the rules at issue in *Morrissey*, both describe how the system functions and are necessary to bring about that functioning. Given the legislative admonition 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 the copyright law,"[125] the Third Circuit should have accepted Franklin's contention that the operating system at issue was the specific system developed by Apple, the system that ran Apple-compatible software. Then, and only then, could the question arise whether and to what extent Apple's copyright in its programs should be limited to allow others to use the system. The Third Circuit made no finding as to whether there was a substantial merger of idea and expression in Apple's programs, leaving that for determination by the trial court on remand.[126]

In the case of computer programs, especially operating system programs, the question of how many ways a given system can be written is particularly difficult. Operating system programs are designed in terms of the physical organization—the architecture—of the computer. They are the interface between the computer and the applications programs which run on the computer to make it perform specific tasks for the computer user. Though there are many different operating system programs which can be written for a given architecture, any given applications program will be designed to run with one, and usually only one, operating system. While Apple presented testimony that an operating system could be produced which would be ninety-eight percent compati-

---

tained solely by the employment of language which gives expression to that which is disclosed.

*Id.* at 184.

122. 379 F.2d 675 (1st Cir. 1967).

123. *See id.* at 678-79.

124. HOUSE REPORT, *supra* note 6, at 5667.

125. *Id.* at 5670.

126. *Apple v. Franklin* was settled before further court proceedings. As part of the settlement agreement, Franklin promised to develop its own operating system programs to run Apple-compatible software. *See* 2 COMPUTER L. REP. 553 (1984).

ble with Apple-compatible software,[127] the difference between ninety-eight percent and one hundred percent compatibility may be enormous when the reliability of software programs run on the system is at issue.[128] It may be that nothing short of one hundred percent compatibility is acceptable within the industry, and that such compatibility is attainable only by wholly or substantially copying the programs developed, in this instance, by Apple.

Some might argue that the rule of *Baker v. Selden* has been taken too far. All that *Baker* says is that the copyright owner cannot prevent the *use* of the system described in the copyrighted work.[129] The operative question should be *who* can use the system, not what system can be used. Under the interpretation of *Baker*, Selden could prevent everyone from using his system except legitimate purchasers of his book. They had paid the royalty price for access to his system, and he could not legitimately prevent them from reproducing his T-account forms for their own use, since to do so would render their investment in his book worthless. Under this interpretation, the vendor and copyright owner of a computer program could not prevent the legitimate purchaser of a copy of the program from using it in his computer, even if such use required the user to copy the program in manner that would otherwise violate the copyright owner's monopoly rights. Section 117, introduced by the 1980 Act, explicitly allows the owner of a copy of a program to make a copy or adaptation of the program if it is "an essential step in the utilization of the computer program in conjunction with a machine."[130]

But the rule of *Baker v. Selden* cannot be so narrowly confined. The *Baker* court itself made no distinction between legitimate owners of copies of Selden's book and others who could be rightfully excluded from using the system embodied in his T-account forms. On the contrary, the Supreme Court explicitly held that the "copyright of a book on bookkeeping cannot secure the exclusive right to make, sell, and use

---

127. Apple Computer, Inc. v. Formula Int'l, Inc., 562 F. Supp. 775, 782 (C.D. Cal. 1983).

128. Franklin included the following testimony in its brief before the Third Circuit:

On rebuttal, Mr. Sander [Apple's witness] asserted that it would be possible to rewrite the Autostart ROM program, for example. However, he conceded that after such rewriting, the computer would *not* be able to run *all* of the Apple-compatible software and achieve total compatibility. As Franklin's Mr. Borden pointed out, reaching less than 100 percent of the market is like making a roof that keeps a house dry less than 100 percent of the time. Obviously, that is not a viable alternative, particularly since it would be impossible to predict which of the thousands of Apple-compatible application programs would be unable to run on an alternate machine.

Brief for Appellee at 20, Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240 (3d Cir. 1983), *reprinted in* 1 COMPUTER L. REP. 681, 695 (1983)(citations omitted).

129. M. NIMMER, *supra* note 105, at § 2.18[C].

130. 17 U.S.C. § 117(1) (1982).

account-books prepared upon the plan set forth in such book."[131] Likewise, in *Morrissey*, the alleged infringer was not a good faith purchaser. The defendant in that case had rejected plaintiff's offer to sell the rights to use his copyrighted sweepstakes rules.[132] Under *Baker* and its progeny, anyone can use a system, once published, and anyone can copy an author's expression if that expression is necessary in order to utilize the system.

But what of Apple's claim that applications programs and operating system programs are indistinguishable, and that any limitation on the copyrightability of operating system programs would prevent effective copyright protection for all computer programs? Is it possible to distinguish applications programs so that they can be protected under copyright law as many believe they should?

Applications programs, like operating system programs, are utilitarian. They allow the computer user to perform a specific task. For instance, a user might choose a spreadsheet program to do her accounting and record keeping or a graphics program with which to produce a picture, design, or animation. Applying the same two-step analysis as is used with operating system programs, one must first determine the application program's function or use. In many cases, the program produces output which itself is copyrightable, such as an audiovisual display, literary work, or it may produce a utilitarian work like a spreadsheet or word-processing system. The second step is to determine how many different programs could be written to produce the same output. Frequently, the answer will be many, many programs. Unlike operating system programs, which may be constrained by both the computer's complex architecture and by the software already developed to run on the pre-existing operating system, applications programs are constrained only by the existing languages in which they can be written and the specific results one wishes to achieve. It is easier to independently create a program that produces a specific result when that result is a visible, user-interactive output than when the result is the coordination of existing software and the computer hardware, a task which requires the careful timing of electrical pulses as they pass through the computer circuitry. To achieve the latter, it might be necessary to study the existing operating system programs to discover "how it was done" in order to develop one's own solution to the problem.[133] Thus there is far less opportunity for an applications programs to monopolize a given re-

---

131. 101 U.S. at 104.

132. 379 F.2d at 677.

133. Because an operating system program controls the operations of the computer and the interconnections between the CPU and peripheral devices, the programmer must trace and coordinate each electrical signal as it travels to and from the CPU. To replicate such signals precisely enough to run software developed for other systems may be impos-

sult than there is for an operating system to do so. Apple's fears that all programs would become uncopyrightable if copyright protection for operating system programs were limited seems unfounded.

## CONCLUSION

We have seen that the transformation from source to machine code will not bar copyright protection for the machine code version of a program. However, that machine code version must be capable of communicating the copyright-protected expression to the program user if it is to be protected as a copy of the original program. While, technically speaking, the machine code version "contains" the protected expression, that expression is, practically speaking, unintelligible. As a matter of policy, the source code should be made available to purchasers of programs in order that they may have access to the copyrighted expression and the ability to modify or adapt the program to their needs as provided by section 117 of the 1980 Act.

Nor does the fact that computer programs are useful articles bar them from copyright protection. That they are utilitarian in nature only subjects them to further scrutiny to insure that the copyright in the program itself cannot be used to create a monopoly in the system, process, or method of operation defined by the program. Where programs perform useful functions, it is the scope, not the fact, of copyright protection which is at issue.

There is yet another aspect of computer programs that deserves attention. The copyrightable expression in a program extends beyond the exact program code to the subroutines, procedures, and other functional elements adopted by the programmer. It is possible that if we allow the programmer to obtain copyright protection for these elements of his program, we may be creating a monopoly in the basic techniques and procedures of program construction, somewhat like allowing mathematicians to copyright particular equations or a painter to protect specific brush techniques. The fact that copyright protects only copying and not independent creation provides little solace for programmers and the development of programming science if courts are willing to find infringement on the basis of access and unconscious, innocent imitation.[134] If the programmer is to be given some copyright protection for the component elements of the program, then, at the very least, broad allowance

---

sible without close study of and substantial copying of the existing operating system. *See* P. HOROWITZ & W. HILL, THE ART OF ELECTRONICS 453-514 (1980).

134. *See* Fred Fisher, Inc. v. Dillingham, 298 F. 145, 148 (S.D.N.Y. 1924) (defendant found guilty of infringement even though he "unconsciously copied" the prior work). The 1976 Copyright Act establishes different levels of statutory damages depending on whether infringement is "willful" or "innocent." 17 U.S.C. § 504(c)(2) (1982).

should be made for the copying and replication of specific elements, and infringement should be found only if the copying involves a substantial number of such elements in a similar sequence designed to bring about the same results. That, and that alone, could protect the art of programming from atrophy due to the monopolization of the basic procedures and other processes that make up most if not all programs.

The computer program is radically different from traditional copyright subject matter. We are only beginning to recognize the extent and implications of this difference. We will have to work out new approaches to this medium of expression so that, while providing the incentives called for under copyright policy, we do not allow authors to establish monopolies over the utilitarian aspects of their programs, either the results produced by the program or the procedures and methods employed by the programmer in writing the program. Because programs differ in their intended results and in their internal structure, not all programs are created equal under copyright law. Some programs produce utilitarian output while others produce output which is independently copyrightable. The former require greater scrutiny than the latter, lest copyright preempt specific methods, systems, or processes. Whether the program results in useful work or copyrightable expression, the nature of the program and the author's expression in it must be carefully considered to prevent copyright from restricting the techniques and methods of programming itself. Only by carefully balancing the interest of the public and the interests of copyright proprietors will we be able to promote computer programs as a new useful art and science.