UIC John Marshall Journal of Information Technology & Privacy Law

Volume 3
Issue 1 *Computer/Law Journal - 1981*

Article 1

1981

Another Look At Copyright Protection of Software: Did the 1980 Act Do Anything For Object Code?, 3 Computer L.J. 1 (1981)

Richard H. Stern

Recommended Citation

Richard H. Stern, Another Look At Copyright Protection of Software: Did the 1980 Act Do Anything For Object Code?, 3 Computer L.J. 1 (1981)

ANOTHER LOOK AT COPYRIGHT PROTECTION OF SOFTWARE: DID THE 1980 ACT DO ANYTHING FOR OBJECT CODE?*

by RICHARD H. STERN†

Available legal protection for mass-marketed computer software is uncertain and unsatisfactory. In particular, protection of software in machine-readable form, such as read-only memories, tapes, and disks, is very insecure. This article discusses the availability of copyright protection for such software, which is termed "object code" in computer parlance, under present law.

The conclusion is that object code is probably not protected by copyright law under the Copyright Revision Act of 1976, as amended in 1980.[1] The present article is directed to the copyrightability[2] of object code under the present law,[3] and provides the foundation for a subsequent article proposing *sui generis* protection of software, particularly object code and algorithms.

† Partner, Baker & Hostetler, Washington, D.C. Chief, Intellectual Property Section, Antitrust Division, United States Department of Justice, 1970-78.

[1] 17 U.S.C. § 101 (Supp. 1980), *as amended by* The Computer Software Copyright Act of 1980, Pub. L. 96-517, 94 Stat. 3015.

[2] "Copyrightability" is used as a shorthand expression for "potentially subject to protection under the copyright laws." *See* text accompanying note 12 *infra*.

[3] The availability of meaningful patent protection for software, even after Diamond v. Diehr, — U.S. —, 101 S. Ct. 1048 (1981), is questionable. Protection under confidentiality agreements is also of questionable benefit for mass-marketed software. *See, e.g.*, Merchant Suppliers Paper Co. v. Photo-Marker Corp., 29 A.D.2d 94, 285 N.Y.S.2d 932 (1967). *But see* Management Serv. Am., Inc. v. Cyborg Sys., Inc., 6 COMPUTER L. SERV. REP. 921 (N.D. Ill. 1978) (summary judgment denied against licensor with six hundred licensees subject to secrecy agreements). The concept of software is uniformly deemed copyrightable. *See, e.g.*, NATIONAL COMM'N ON NEW TECHNOLOGICAL WORKS, FINAL REPORT 20 (1978) [hereinafter cited as CONTU REP.]; Cary, *Copyright Registration and Computer Programs*, 11 BULL. COPYRIGHT SOC'Y 362 (1964).

Although many of these issues are outside the scope of this article, they serve to illustrate the present inadequate state of legal protection for software.

This article presupposes a general familiarity with copyright law and computers. Some definitions need to be agreed upon, however, and some brief discussion of the technology involved is appropriate.

Program. The term "program" will be used to refer to a particular procedure for operating a digital computer or a machine system incorporating a digital computer to accomplish a particular result.[4]

Source Programs. The term "source program" will be used to mean a program written in a high level computer programming language such as FORTRAN, BASIC, or COBOL. A statement in such a language is said to be in source code. A source program is a series of formalized statements (instructions) for operation of the computer, stating that a series of things are to be done to data made available to the computer.[5] The term "high level" means that such languages as BASIC are close to ordinary English. Indeed, BASIC may be regarded as a dialect of English.[6]

Object Code. The terms "object code" or "object program" will be used to refer to a mechanical counterpart of a source program. Object code is directly usable in a machine, and is not written at all (in the ordinary sense of that word), but is embodied in magnetic tape, disks, or other physical device, such as a read-only memory (ROM).[7]

Object code does not correspond to the source code to which it

[4] A computer program is defined by statute as "a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101 (Supp. 1980). In WORLD INTELLECTUAL PROPERTY ORGANIZATION, MODEL PROVISIONS ON THE PROTECTION OF COMPUTER SOFTWARE 9-11 (1978) [hereinafter cited as WIPO, MODEL PROVISIONS], "computer program" was defined as "a set of instructions capable, when incorporated in a machine-readable medium, of causing a machine having information-processing capabilities to indicate, perform, or achieve a particular function, task, or result."

[5] An example of a typical BASIC instruction to the user of the computer "INPUT X," which instructs the computer to print out a *?* on the display, whereupon the user types the correct value of variable X on the keyboard. Such an instruction often may be preceded by an instruction to the computer such as "PRINT 'WHAT IS RADIUS, IN INCHES'," which will cause the computer to print out "WHAT IS RADIUS, IN INCHES" before the *?* and to treat whatever number the user types after the *?* as the value of X. Thus, PRINT X may be considered an instruction to the computer to print out an instruction to the user.

Other typical BASIC instructions to the computer can be of the form "GOTO 130," meaning that the computer should carry out the instruction on line 130 of the program; "IF I10, THEN GOTO 130," meaning if the value of variable I is less than 10, the computer should carry out the instruction on line 130, but otherwise it should do the next listed step; of "PRINT 'ANSWER IS'; X+Y," meaning that the computer is to calculate the sum of variables X and Y and print out the message "ANSWER IS ——," with the calculated sum printed in the blank.

[6] See note 5 *supra*.

[7] Source code can also be embodied in a tape or other device, and need not be

is related on a simple and straight-forward basis. Object code is not simply another dialect of English. Nor is it a translation of source code in the sense that one translates a book from French into English, or even translates a source program from FORTRAN to BASIC.[8] In fact, object code is superficially unintelligible, because it appears as a meaningless sequence of electronic signals of "0" and "1" value.[9] An object program is a collection of object code which will directly enable the computer to execute the corresponding source code program.

Compilation. The transformation of source code into object code may be accomplished in various ways. Typically, it is done in one or more stages in which the set of statements progresses from a high level language, like BASIC, through one or more successively lower language levels which are more and more removed from ordinary or natural English and less and less intelligible.[10] The process, sometimes termed "compilation," is accomplished by a computer and appropriate programming. A separate compiler program will be needed for a particular high level language, like FORTRAN, processor, and program storage device, like a particular model of ROM. Different compiler programs are written for each such combination.[11]

Typically, in the course of compilation, each source code statement will be replaced by one or more lower level statements. In between these lower level statements, "linking statements" may be inserted. Some statements in the source code version may disappear. The ultimate product of the process will be a set of 1/0 signal

printed on paper to be retrieved and understood. Source code is not directly usable in a machine, for it must first be transformed into a machine-usable language.

[8] Object code has syntax rules that require a program in object code to contain many more statements than does the program in source language. There is no simple 1:1 or 1:5 relationship between the number of source code and object code statements. In transforming source code to object code, some source code statements may wholly disappear, while some object code statements may have no counterpart in the source code version. See text accompanying note 11 *infra*.

[9] *See* Data Cash Sys., Inc. v. JS&A Group, Inc., 480 F. Supp. 1063, 1065 (N.D. Ill. 1979), *aff'd on other grounds*, 628 F.2d 1038 (7th Cir. 1980).

[10] Two very common methods are by use of "compiler" and "assembler" programs. A compiler program transforms a program in source code into object code, so that as far as the user is concerned the input is source code and the output is object code. This does not mean that the computer accomplishes this transformation in a single pass. An assembler program transforms a program from assembly code to object code, meaning that the program has to be written in assembly code in the first place or transformed from source code to assembly code. Assembly language is a lower level language than source code. Particular assembly code versions of a single source program will be specific for different types of computer.

[11] CONTU REP., *supra* note 3, at 21 n.106.

level sequences, the object code. These may be stored on a tape or other memory storage device. If the device using the program is self-contained, the storage medium will often be a ROM, a postage-stamp sized device containing thousands of chemically printed switches. A pattern of open and closed switches (0's and 1's) embodies the object code. If a large, multi-purpose computer is involved, the storage medium is more likely to be tapes or disks. A pattern of unmagnetized and magnetized zones embodies the object code on tapes and disks.

I. COPYRIGHTABILITY OF SOFTWARE

The term "copyrightability" will be used as a shorthand expression for "able to be protected under the copyright laws." That is, a software item is copyrightable if copyright protection can be obtained for the item. Intrinsic to this protection is the right to successfully bring an action for copyright infringement against one who duplicates or commercially uses the item.

Source programs and documentation relating to them are generally recognized to be copyrightable.[12] They are written in English or a language readable by human beings; they contain intelligible messages; they are generally agreed to be writings or works of authorship[13]; and they are accepted for registration in the Copyright Office as literary works.[14]

[12] Literary works, as defined in the Copyright Act, include works expressed in words, numbers, or other verbal or numerical symbols, in books, tapes, discs, or cards. 17 U.S.C. § 101 (Supp. 1980); H.R. REP. No. 1496, 94th Cong., 2d Sess. 54 (1976) (literary works include computer programs.); S. REP. No. 473, 94th Cong., 1st Sess. (1975) (computer programs are copyrightable as literary works). The Computer Software Copyright Act of 1980 states that source programs are copyrightable.

[13] The United States Constitution authorizes Congress to pass laws to secure to authors a limited exclusive right to their writings. Art. I, § 8, cl. 8. The United States copyright laws no longer use the term "writings" to describe what is protected under copyright law. Instead, they now use the term "original works of authorship" (17 U.S.C. § 102 (1979)), but the Constitution nonetheless limits the scope of the statute to "writings." *See* Graham v. John Deere Co., 383 U.S. 1, 5 (1966) (discussing the same clause of the Constitution from the patent standpoint). A non-writing cannot be copyrighted. *See* The Trademark Cases, 100 U.S. 82 (1879). In that decision, the United States Supreme Court held that Congress cannot regulate trademarks under the Patent-Copyright Clause. As a result, it was necessary to pass a new trademark law based on the Commerce Clause of the Constitution. *See* 15 U.S.C. §§ 1051-1150 (1946).

[14] Since May 1964, the Copyright Office has accepted source programs for deposit and registration as literary works. UNITED STATES COPYRIGHT OFFICE, COPYRIGHT REGISTRATION FOR COMPUTER PROGRAMS (1964) *reprinted in* 11 BULL. COPYRIGHT SOC'Y 361 (1964); Cary, *Copyright Registration and Computer Programs*, 11 BULL. COPYRIGHT SOC'Y 362 (1964). In the announcement, the Copyright Office in-

On the other hand, it is clear that a machine system utilizing particular software is not copyrightable under United States law,[15] even if the software *per se* were copyrightable. Similarly, a programmed memory device is not copyrightable as such. Whether the object code "message" within a programmed memory device is copyrightable as such is a difficult and highly controversial question.

The phrase "copyrightable as such" has been used here, because a potential distinction exists between (1) having a copyright in *X* and protecting *X* under the copyright laws, and (2) having a copyright in *Y* and protecting *X* because of its relationship to *Y*. The conclusion that object code is not copyrightable as such does not require the conclusion that duplication of object code cannot be the infringement of some other copyrightable writing, such as a source program. However, these two decisions are actually linked.

II. SCOPE OF COPYRIGHT PROTECTION: THE PROBLEM WITH OBJECT CODE

The scope of protection afforded copyright owners is set forth in the current copyright law. Section 106 provides that the copyright owner has the exclusive rights to do or authorize the following things, among others:

(1) to reproduce the copyrighted work "in copies";

(2) to prepare "derivative works based upon" the copyrighted work; and

(3) to distribute "copies" of the copyrighted work, by sale,

dicated some doubt as to whether a program as such is the "writing of an author" and whether object code is a "copy" that can be registered. The Office decided to resolve doubt in favor of registrability. *Id.* This action facilitates access to the courts for proprietors of software, and permits the courts to resolve the copyrightability issue. The Office determined that copies of programs in human-intelligible form would be registered, but when publication or dissemination "was in a form that cannot be perceived visually or read," a printout would have to be deposited for registration. *Id.*

Most authorities now assume that source programs are protected by copyright regardless of the form of the coding medium. *But see* Pope & Pope, *Protection of Proprietary Interests in Computer Software*, 30 ALA. L. REV. 527, 546 (1979) (duplication of punched cards or magnetic tape containing source code is not the making of an "infringing copy" of the source program).

[15] Machines are not copyrightable. A machine is not a "writing." *See* note 13 *supra*. *See also* Muller v. Triborough Bridge Auth., 43 F. Supp. 298 (S.D.N.Y. 1942) (copyright protection of drawing not extended to the depicted bridge). The copyright law itself denies copyright protection to utilitarian aspects of articles. *See* 17 U.S.C. §§ 101, 102(b) (1979); Esquire, Inc. v. Ringer, 591 F.2d 796 (D.C. Cir. 1978). By contrast, under United Kingdom law, a device may be protected, at least indirectly, by a copyright on a blueprint. The competitive device is considered a copy of the blueprint. *See, e.g.*, L.B. Plastics Ltd. v. Swish Prod. Ltd., [1979] Fleet St. Rep. 45 (H.L.).

rental, lending, etc.[16]

Unless the conduct of the alleged infringer falls within one of these categories, the owner cannot obtain relief under the copyright law.

The severe problems encountered in trying to secure copyright relief against appropriation of object code are illustrated by the trial court's decision in *Data Cash Systems, Inc. v. JS&A Group, Inc.*,[17] decided under the Copyright Act. Data Cash had hired a programming consultant to develop a computer program for use in a handheld calculator chess game known as "CompuChess." The program was stored as object code in a ROM incorporated into Data Cash's product. About a year after Data Cash began to market CompuChess, JS&A entered the market with an almost identical product. Taking advantage of recent technological advances, JS&A apparently unloaded the ROM of Data Cash's device, so that, in effect, the object code was directly transferred to the ROM of the JS&A device.[18]

Data Cash was unable to secure copyright relief against the appropriation of its program for several reasons. Most significantly, JS&A had not made a copy of Data Cash's copyrighted source program. A copy of a source program is a source program, according to the trial court, not object code or a mechanical device, such as a programmed ROM. The court said, "[t]he ROM is not a 'copy' of the plaintiff's computer program and therefore the [so-called] copying is not actionable."[19] Although the court's ruling was made under the 1909 Copyright Act, which is superseded by the present Act, the court indicated that for constitutional reasons it would have reached the same result under the 1976 Act.[20]

The Seventh Circuit affirmed on wholly different grounds, refusing to pass on the issue decided below. Data Cash had marketed several thousand copies of the machine, including the ROM, without any copyright notice. It had done so because it mistakenly believed that a ROM could not be unloaded. So extensive a publication without proper copyright notice, the court held, worked a forfeiture of

[16] 17 U.S.C. § 106 (1979).

[17] 480 F. Supp. 1063 (N.D. Ill. 1979), *aff'd on other grounds*, 628 F.2d 1038 (7th Cir. 1980).

[18] The *Data Cash* court stated that it did not know how the ROM in defendant's product was created. 480 F. Supp. at 1071. The parties had stipulated that defendant's supplier manufactured the ROM from a punched tape received from a Hong Kong company. The parties speculated that someone unloaded plaintiff's ROM by decoding the object program by means of a computer and then either printing it out or placing the unloaded signals into another ROM. Either way, ROMs could then be made that duplicated plaintiff's ROM. *Id.* at 1071 n.14.

[19] *Id.* at 1069.

[20] *Id.* at 1066-67 n.4.

any copyright Data Cash had.[21] The decision may hold by implication that if Data Cash had used a proper notice it would have secured copyright protection for the ROM's object code, but this is a highly speculative interpretation when the opinion does not address the question directly.

The facts of the *Data Cash* case also suggest a jurisdictional problem. In some cases, the copying and sale of the copy may take place only outside of the United States. Thus, an unauthorized copyist in Hong Kong might acquire a source program. The source program could then be compiled in Hong Kong as object code and embodied in machines in Hong Kong or in the United States. The machines could then be sold and used in the United States. Under this set of facts, there would apparently be no actionable copyright infringement under United States law.

This highlights a basic problem when United States copyright law is used to protect software: commercial use of software is not copyright infringement, because use, or execution of a program, does not constitute copying.[22] This short fall is one of the fundamental defects in copyright law as a means of protecting software in the United States.[23]

[21] 628 F.2d at 1041-43. The court of appeals indicated that opposing counsel below had not briefed the copy issue and were not interested in pursuing it on appeal. *Id.* at 1041.

[22] Commercial use of concrete ideas set forth in literary works is not a copyright violation. Baker v. Selden, 101 U.S. 99 (1879). Thus, under United States law, it is not a copyright infringement to bake a cake in accordance with a recipe printed in a cookbook, nor to knit and sell a sweater described in a knitting instruction book.

Conceivably, inputting a source program to a computer equipped with an appropriate compiler program might be copying the program but this is doubtful. *See* CONTU REP, *supra* note 3, at 22. On the other hand, use of a LIST command, causing the computer's peripheral equipment to print a human-readable copy, would be the making of a copy. In any event, execution of the program is not in itself making a copy any more than baking the cake is making a copy of the recipe. *See, e.g.*, De Silva Constr. Corp. v. Herrald, 213 F. Supp. 184 (N.D. Fla. 1962); Muller v. Triborough Bridge Auth., 43 F. Supp. 298 (S.D.N.Y. 1942). *But see* L.B. Plastics Ltd. v. Swish Products Ltd., [1979] Fleet St. Rep. 145 (H.L.).

[23] United States law may afford protection against imports in situations like that of the *Data Cash* case, under the Tariff Act of 1930. 19 U.S.C. § 1337 (1980). That law prohibits unfair methods of competition in import commerce and directs the United States International Trade Commission to prohibit such conduct when it threatens substantial injury to an economically and efficiently operated domestic industry. Infringement of a patent, trademark, or copyright can be prevented under this statute. Moreover, the statute's concept of unfair is not limited to conduct prohibited by the patent, copyright, and analogous laws. Proceeding before the ITC may lead to a ruling on whether importation of ROMs unloaded from a domestic complainant's ROMs is an unfair method of competition, on the theory that such conduct violates the spirit of the United States copyright laws. Accordingly, even though no copyright relief

III. THE EFFECT OF THE 1980 SOFTWARE COPYRIGHT ACT

Federal legislation, called the Software Copyright Act of 1980, amended the copyright laws concerning software. In the 1976 general revision of the copyright law, Congress was unable to agree on the proper scope or application of copyright law to computers. Accordingly, Congress legislated, in section 117 of the 1976 Act, that the state of the law on copyrightability of computer material would be preserved as it was on December 31, 1977.[24] At the same time, Congress formed a special Commission to make recommendations for copyright legislation on various computer-related matters.[25]

Congress enacted the recommended legislation several years later.[26] The new law overtly did two things:

> (1) It defined "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."[27]
>
> (2) It amended section 117 to state that it is *not* an infringement for the owner of a copy of a computer program to make copies or adaptations needed in utilizing the program or copies for archival purposes.[28]

Some commentators have viewed this legislation as dramatically broadening copyright protection on software.[29] The declaration that certain acts with respect to programs are *not* copyright infringements is regarded as a statement, by implication, that any other generally similar acts *are* infringements. Some commentators are so enthusiastic that they regard the language as a legislative rever-

against appropriation of object code is available domestically, it may be possible for software proprietors to prevent importation of devices embodying copied object code.

[24] Copyright Act of 1976, Pub. L. No. 94-553, Title I, § 401, 90 Stat. 2541 (formerly codified as 17 U.S.C. § 117, but now superseded). This section provided that owners of copyrights were to have no "greater or lesser rights with respect to the use of the work" in computers than the rights "afforded to works [sic] under the law . . . in effect on December 31, 1977 . . . ." The decision not to legislate on computers was made about eight years earlier. *See* CONTU REP., *supra* note 3, at 39 n.163.

[25] Act of December 31, 1974, Pub. L. No. 93-573, 88 Stat. 1873, *as amended by* Act of October 28, 1977, Pub. L. No. 95-146, 91 Stat. 1226.

[26] Copyright Software Act of 1980, Pub. L. No. 96-517, 94 Stat. 3015. The statute primarily concerns government ownership of patents and administrative reexamination of patents.

[27] *Id.*, *amending* 17 U.S.C. § 101 (Supp. 1980).

[28] *Id.*, *amending* 17 U.S.C. § 117 (Supp. 1980).

[29] *See, e.g.*, *For Better or Worse*, DATAMATION, Feb. 1981, at 49; *Program Writers Get Expanded Protection*, COMPUTER CAREER NEWS, Jan. 12, 1981, at 3. Greguras, *Developments in the Law of Copyright Protection of Computer Software*, in UNIVERSITY OF SOUTHERN CALIFORNIA SECOND COMPUTER LAW INSTITUTE 93 (1981).

sal of the *Data Cash* decision's denial of protection to object code.[30] The premise of this reasoning is that the other parts of the 1976 Copyright Revision Act would have already accomplished these results, but for the now deleted section 117 that sought to freeze the law where it was.[31] According to this view, the removal of section 117 operated like a discovered check in chess, unleashing and permitting full operation now of a previously partly veiled or curbed statutory scheme.[32]

Whether the 1980 amendment does any such thing necessarily depends on whether the 1976 copyright law changed anything relating to object code software. First, did the 1976 law make object code copyrightable as such, without reference to source programs or anything else? If not, did it prohibit the unloading, duplication, and marketing of object code as impermissible infringement of any underlying work embodied in a copyrighted source program? Under the old copyright law, these would be different questions with independent answers. Under the new law, it appears that these are merely different formulations of whether object code is a copy of a work of authorship.

The only decision to analyze the copyright status of object code is the trial court's opinion in *Data Cash Systems, Inc. v. JS&A Group, Inc.* It suggests that object code is not a writing and was thus statutorily uncopyrightable under the old law and constitutionally uncopyrightable under the new.[33] One must therefore turn to a textual analysis of the new copyright law and to the Constitution.

As already indicated, there are several possible ways in which object code could be protected by the copyright laws:

> (1) Object code could be deemed a "copy" of a source program, and thus an infringement of it. This is the most important possible route to protection.
>
> (2) Object code could be copyrightable as such. Copying it would then be an infringement.
>
> (3) An object code counterpart of a copyrighted source program could be deemed a derivative work based on the source program. Preparation of a derivative work is an infringement of the underlying work.[34]

[30] *See* text accompanying notes 17-21 *supra*.

[31] *See* note 24 *supra*.

[32] *See, e.g.*, CONTU REP., *supra* note 3, at 12-13.

[33] 628 F.2d 1041-43 (7th Cir. 1980).

[34] *See* text accompanying notes 12-16, *supra*.

A. Copying Source Programs

"Copies" are partially defined in the statute as material objects in which a work is fixed for the first or a subsequent time.[35] The work must be capable of being "perceived, reproduced, or otherwise communicated" from the copy, either directly or "with the aid of a machine."[36] In one view, this definition implies that object code is a copy of the source program. "Because works in computer storage may be repeatedly reproduced, they are fixed and, therefore, are copies."[37]

A second and opposing view is that the statutory requirement that the work be capable of being perceived or *otherwise* communicated from the alleged copy requires human perception rather than mere mechanical registration of a message with some intellectual content.[38] In this view, object code, unintelligible in itself, is not a "copy" of a programmer's expression of his programming idea, any more than a cake is not an intelligible copy of the recipe writer's idea and a machine is not an intelligible copy of the designer's drawing.[39] Indeed, object code is even less intelligible than a cake or machine. The latter do not communicate an intelligible message; but are directly perceived by the senses. Object code itself communicates nothing to human senses.

This objection to the copyrightability of object code has nothing

[35] 17 U.S.C. § 101 (Supp. 1980).

[36] *Id.* Whether a machine may be used to aid in perception would not seem significant. The legislative history indicates that the reference to aid of a machine was intended to permit the copyrighting of phonograph records, videotapes, and similar devices which permit a listener or viewer to perceive message with the assistance of a phonograph, TV, or the like.

[37] CONTU Rep., *supra* note 3, at 22. The observation may mean that ROMs (or other embodiments of object code) are commercially mass-produced in identical units, and therefore, are considered to be "copies." This begs the question, however, for a million Chevrolet radiators can be mass-produced from a blueprint without their "copies" of the blueprint or of any copyrightable work. *See* note 15 *supra*. Indeed, the radiators may be substantially similar to one another, the prototype, and the blueprint, but they are nonetheless not "copies" of any of them.

[38] CONTU Rep., *supra* note 3, at 32 (dissenting views of Commissioner Hersey). Initially the Commission considered recomending that (1) object code has no more "lack of communicative potential" than a book written in Sanscrit or a table of trigonometric functions, which are closely copyrightable, and (2) if lines must be drawn between copyrightable and uncopyrightable forms of programs, the judiciary should be assigned the task, on a case-by-case basis. *Id.* at 10. The Report's final version states only the second of these positions. *Id.* at 22-23. The general flavor of the CONTU Report is in favor of protection of object code, but the specific language of the Report is unequivocal statements of what the copyright law is or recommendations that it be changed to embrace object code.

[39] *See* note 15 *supra*.

to do with its embodiment in an unconventional literary medium, such as tape, disk, or ROM. In the first place, source code may also be so embodied, and it may be called forth in an intelligible form by the command PRINT. Audiovisual material may be in the form of videotape. Moreover, the legislative history indicates that copies may be made in any kind of recording medium, so long as the other provisions of law are satisfied. Therefore, object code, when it produces an intelligible display, fixes a message, just as videotape may fix an opera or motion picture.[40]

The object code counterpart of a source program, however, differs significantly from a cassette or videotape in which a work such as *Don Giovanni* or *Casablanca* is fixed. Playing the tape of a movie or opera always results in the same output: Leporello will always have 1003 entries for Spain in the Don's catalog, never 999 or 1004; Bogart will always leave Casablanca at the end to join the Free French Army, and he will never get on the airplane for Lisbon. In contrast, the output display caused by a program on a disk or ROM will never be the same twice unless the same data input is used. Because the result is a function of the data processed, object code's output cannot be considered a copy of any preexisting work. The display produced by the object code cannot be the work embodied in the object code. That work can only be the program itself. Object code, however, is not itself intelligible, for a printout of object code is unintelligible even to trained observers.[41]

On balance, the statutory definition of copy would seem to be conditioned on human-intelligibility. Object code is *not* a copy of anything that is itself copyrightable.

A counter-argument is that object code does embody a work (the source program) in a form from which the work can be perceived or otherwise communicated with the aid of a properly programmed computer. That is, the original compiler program used to compile the object code from the source program, and the other relevant machine parameters, could in theory be used to devise a computer program that would reverse compile object code embodied in a ROM, disk, or the like, transforming it into intelligible source code. The resulting source code would probably be close enough to the original source program to constitute a copy of it, although the testimony of an expert witness may be necessary to establish sub-

[40] *Accord*, CONTU REP., *supra* note 3, at 27 (Nimmer, Comm'r, concurring).

[41] *See* note 9 and accompanying text *supra*. A different view may be suggested by a recent trial court opinion in which the court granted a preliminary injunction against a pirated video game on the theory that the game's audiovisual display was copyrighted even though the stored program was not. Stern Elec., Inc. v. Kaufman, No. 80-03248 (E.D.N.Y., May 20, 1981).

stantial similarity. For this reason, the original work can be derived in a perceivable form—source code—from the object code. Therefore object code is intelligible, not directly, but with the aid of a machine to process it.

This argument is not sound. That comprehensible information can be extracted from a device, like ROM, not intended to function as a medium for human communication, does not make the device itself an object from which the work can be perceived or otherwise communicated. The argument proves too much. For example, a work such as a blueprint of a chair could similarly be reverse-compiled from the actual chair by photographic means. But that does not make the chair a copy of the work. The law is quite against extending copyright protection to such copies.[42]

The 1980 amendments do not change the definition of copies in the 1978 Act. If anything, the 1980 Act merely reinforces the view that copies of programs must be in intelligible form for them to be protected under the copyright laws. First, it defines computer programs as a set of statements or instructions to be used directly or indirectly in a computer to bring about a result. One might argue that directly must be a reference to object code, since a source program is used in a computer indirectly, because it must be transformed into object code to be used in a computer. This is a very weak argument because there is no legislative explanation to support this interpretation. Moreover, nothing in the 1980 statute states that all computer programs are potentially copyrightable. There is no express reference in the statutes or legislative history to the copyrightability of object code as distinguished from that of computer programs generally.[43]

[42] *See* notes 15, 22 *supra. See also* Jack Adelman, Inc. v. Sonner's & Gordon, Inc., 112 F. Supp. 187 (S.D.N.Y. 1934).

[43] The CONTU Report is equivocal, at times expressing enthusiasm for copyrightability of all forms of programs, whether intelligible or not, and at other times suggesting that the courts should sort it all out later. Moreover, there is a pervasive imprecision in its specific language, suggesting possible apprehension of being caught in an untenable legal position as to copyright law. *See e.g.*, note 37 *supra*. The preponderant tone of the CONTU Report, however, is that in most of its members' view the 1976 Act had already allowed copyright in object code, so that all that was needed to protect object code was to unleash the 1976 Act. The CONTU Reports legislative recommendations appear to be formed on this premise, and Congress did eventually enact the recommended legislation three years later.

It may therefore be argued that Congress rubber-stamped the CONTU Report majority views by enacting the 1980 amendments. The difficulty with this approach is that its basic premise is that the 1977 CONTU Report, interpreting the 1976 Act, is codified by 1980 legislation that is itself silent on the subject. Such legislative history by delegation is always highly questionable. When it involves an implied rewrite of an earlier-passed statute, without Congress' having used express language directed to

Secondly, the 1980 amendment deleted former section 117 and substituted a new one. The new section declared that it is *not* an infringement for a rightful owner of a program to adapt the program. The report to Congress recommending enactment of section 117 gave two examples of adaptation of programs.[44] One example was conversion of a program from one higher-level language to another, perhaps from BASIC to FORTRAN.[45] This is clearly a reference to source programs, for only they are written in high level languages.[46] The other example was to add features to the program. Again, this necessarily involves source code. Adaptation is a procedure that can be done only at the source code level. It calls for a human being to revise a program intelligible to him, by modifying an intelligible copy written in source code, not object code.

B. Direct Copyrightability

Object code is not subject to copyright directly as a work of authorship. A work is created under the new Act when it is fixed for the first time in a copy.[47] A work is fixed when it is embodied in a copy from which it can be perceived or otherwise communicated.[48] Since object code cannot be perceived by or communicated to a human being, it is not a copy and thus not subject to copyright.[49] In addition, object code does not fit within any of the seven classes of copyrightable works set forth in section 102(a).[50]

that end, it is doubly questionable. Reliance on the extra-Congressional 1977 CONTU Report for interpretation of the Congress' 1976 Act would therefore seem to be unjustified.

[44] CONTU Rep., *supra* note 3, at 13.

[45] *See* Synercom Technology, Inc. v. University Computing Co., 462 F. Supp. 1003, 1013 n.5 (N.D. Tex. 1978) (adaptation of user's manual).

[46] *See* text accompanying notes 4, 5 *supra*.

[47] 17 U.S.C. § 101 (1979).

[48] *Id.* The statute is repetitious when referring to things being perceived or otherwise communicated. The phrase occurs in section 101 in the definitions of copies, fixed, and phonorecords. It also occurs in defining the subject matter of copyright 17 U.S.C. § 102 (1979). Further, the terms "copies" and "fixed" are used to define various other concepts, as well as circularly to define each other.

[49] The better view would be that human intelligibility is a predicate for finding something to be a "copy." *See* text accompanying note 39 *supra*.

[50] Section 102(a) states that works of authorship include the following categories: (1) literary works, (2) musical works, (3) dramatic works, (4) pantomimes and choreographic works, (5) pictorial, graphic, and sculptural works, (6) motion pictures and other audiovisual works, and (7) sound recordings. 17 U.S.C. § 102(a) (Supp. 1980). There may be other types of work which are copyrightabe, but not listed in section 102(a), but there is no suggestion that object code is such a work.

C. Derivative Works

Object code is not a derivative work derived from a source program. A derivative work is a new work of authorship based on, and containing recognizable elements of, an earlier work.[51] Examples are a translation of *Hamlet* into French, an edited version of *Hamlet*, or a dramatization of one of Bocaccio's tales such as *All's Well That Ends Well*. The copyrightable elements of a derivative work are only those elements *not* taken from the earlier work. In other words, only ideas that are newly created by and original with the author of the derivative work may be copyrighted.[52]

These characteristics of a derivative work are not present in object code. First, compiling a source program into object code does not involve even the small amount of creativity and originality required for a derivative work,[53] since such compilation is routinely done by computer. In any event, the same copy problem is present with derivative works as with the other approaches to copyrightability already discussed.[54] Like any other work of authorship, a derivative work based on a source program must be fixed in a copy, which is perceivable by or otherwise communicated to human beings. Object code therefore has no higher status as a derivative work than it does as an original work or as a copy of an original source code work.

D. Utility

The fundamental problem with object code is that it is *not* a medium intended for communication to humans. Object code *is* intended for operating a machine and doing useful work. A traditional principle of copyright law, carried forward in the new Act, is that copyright does not protect useful or utilitarian aspects of works. Copyright law traditionally has distinguished between useful and utilitarian articles, on one side, and articles that communicate information between persons on the other. Chairs, light fixtures,[55]

[51] 17 U.S.C. §§ 101, 103 (Supp. 1980); *see* L. Batlin & Son, Inc. v. Snyder, 536 F.2d 486 (2d Cir.), *cert. denied*, 429 U.S. 857 (1976).

[52] *Id.* (derivative work based on public domain toy bank). *See also* Durham Indus., Inc. v. Tomy Corp., 630 F.2d 905 (2d Cir. 1980) (derivative work based on copyrighted Disney cartoons).

[53] 536 F.2d at 488, 491 (use of physical, rather than artistic, skill to transform work from one medium to another held insufficient for copyrightability).

[54] *See* text accompanying notes 36-39 *supra*.

[55] Thus, a statuette of a Balinese dancer converted to a lamp is copyrightable only as a statuette, not as a lamp. *See generally* Mazer v. Stein, 347 U.S. 201 (1954). If the lamp and statuette ingredients are inseparable, nothing may be copyrighted. Esquire, Inc. v. Ringer, 591 F.2d 796 (D.C. Cir. 1978).

bridges[56] and machines are examples of the former; books, musical scores, maps, and posters, of the latter.

The current copyright law continues this distinction by making useful articles works of authorship only "insofar as their form but not their mechanical or utilitarian aspects are concerned," and only if, and to the extent, that their design incorporates intellectually-apprehended features that can be identified separately from and can exist independently of the article's utilitarian aspects.[57] Similarly, copyright protection is expressly withheld from 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 a work.[58]

When the primary intended use of an article is utilitarian, in the sense that the article itself performs a mechanical or utilitarian function or it interacts with machines, such as computers, to do so, the article would appear to be outside copyright law. The definition of copy supports this view, as do other passages of the copyright law. The article is within the copyright law's protection only when, and to the extent that, its intended use is as a medium for communication. Object code is a medium for storing data, but not intelligible, communicable data. Object code is in no sense a medium of communication, and it is therefore not copyrightable.

This point may need some clarification. Object code in this context means the machine-readable counterpart of source code in a source program. It does *not* mean intelligible data, such as statements in a natural language such as English or French, or in a human-intelligible language such as BASIC, stored on disk, tape, or ROM. A disk, tape, or ROM used to store such data, and from which the data can be retrieved and directly displayed in human-intelligible form, would be analogous to a book or motion picture film.[59] But a disk or ROM used to store a program in machine-readable form, unintelligible to humans, would be a useful article.

The utility/communication dichotomy reflects the two basic pur-

[56] *See* Muller v. Triborough Bridge Auth., 43 F. Supp. 298 (S.D.N.Y. 1942).

[57] 17 U.S.C. § 101 (Supp. 1980).

[58] 17 U.S.C. § 102(b) (1979). *See also* CONTU REP., *supra* note 3, at 28 (Dissenting Commissioner Hersey contends that a computer program "in its mature and usable form," object code, is an uncopyrightable mechanical device).

[59] It is possible to store both program materials and intelligible messages in ROMs although doing the latter is uncommon. A very short animated cartoon could be stored in a ROM. Indeed, coin-operated video game ROMs contain stored material of this type to attract customers. Moreover, short musical tunes are stored on digital watch ROMs, for alarm clock purposes. The output of such ROMs is not varied to respond differently to different particular inputs, but is fixed.

poses and directions of the Constitution's provisions concerning intellectual property, and their implementing status. The Constitution has two branches, tabulated below:

Patent Branch[60]	Copyright Branch[61]
To promote progress of useful arts	To promote progress of science (*scientia*)[62]
By granting patents to inventors	By granting copyright to authors
To protect discoveries	To protect writings
Utilitarian aspects	Communicative aspects.

The implementing statutes embody the same dichotomy. Patents protect new and useful advances in machines, processes, and the like,[63] while copyrights protect original works of authorship.[64]

Object code, as a practical matter, falls between these two statutory schemes. The 1980 software copyright amendments did nothing to remedy this lack of protection. In any case, it is unlikely that legislation based on the copyright clause, rather than the commerce clause, would pass constitutional muster.[65]

CONCLUSION

The copyrightability of object code is doubtful, and there are no definitive judicial rulings. Accordingly, argument can be made for either view. At the present time it would be imprudent to expect more from the new copyright law than prior software decisions have offered. The new law may have covertly made software copyrightable, but this will only be certain after considerable litigation of the issue.

The better view is that to be protected under copyright law, an alleged copy must be intended to communicate some message intelligible to human beings, even if they need a machine to aid in the communication. That an object intended for a utilitarian purpose

[60] "The Congress shall have Power . . . To promote the Progress of . . . useful Arts, by securing for limited Times to . . . Inventors the exclusive Right to their . . . Discoveries." U.S. Const. art. I, § 8 cl. 1, 8.

[61] "The Congress shall have Power . . . To promote the Progress of Science . . . by securing for limited Times to Authors . . . the exclusive Right to their . . . Writings. . . ." *Id.*

[62] Science (*scientia*) is equated to general knowledge. *See generally* Gramham v. John Deere Co., 383 U.S. 1, 5 (1966).

[63] 35 U.S.C. § 101 (Supp. 1980).

[64] 17 U.S.C. § 102(a) (Supp. 1980).

[65] *See* note 13 *supra*.

may be made to disgorge something intelligible to human beings if placed in a proper machine is probably insufficient. By these criteria, object code is not a copy of a work of authorship, and is thus not directly protected by copyright, nor indirectly protected under the theory that it is an infringement of a copyright based on the source program.

Perhaps, Congress will enact a clearer statute, unequivocally declaring that object code is copyrightable but unless and until it does so, duplicaton of object code will probably continue to avoid the United States copyright laws.