SOFTWARE PROTECTION: THE 1985 PERSPECTIVE

David Bender

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

Recommended Citation
SOFTWARE PROTECTION: THE 1985 PERSPECTIVE†

DAVID BENDER*

TABLE OF CONTENTS

I. INTRODUCTION ........................................... 406
II. UNUSUAL ASPECTS OF THE COMPUTER PROGRAM ........... 407
III. THE NATURE OF A COMPUTER PROGRAM .................. 408
IV. SOURCES OF COMPUTER PROGRAMS ......................... 410
V. LEGAL METHODS FOR PROTECTING SOFTWARE ............... 412
   A. Patent Protection of Programs ......................... 412
      1. Nature of Patent Grant .............................. 412
      2. Obtaining a U.S. Patent ............................. 413
      3. Requirements for a U.S. Patent ...................... 413
      4. Statutory Subject Matter ............................. 414
      5. Infringement .................................... 417
      6. Advantages and Disadvantages of Patent Protection 418
   B. Copyright Protection of Software ....................... 419
      1. Statutes ........................................ 419
      2. Copyright Cases Relating to Software Protection 423
      3. The Work Made for Hire Doctrine .............. 432
      4. Advantages and Disadvantages of Copyright Protection 433

† © 1985 David Bender. Earlier versions of portions of this article have appeared in continuing legal education materials.

* Of Counsel, White & Case, New York, NY; Sc.B., Brown University; Graduate Engineering Fellow, Cornell University; L.L.B., University of Pennsylvania; LL.M. in Patent & Trade Regulation Law, George Washington University; S.J.D., George Washington University. Mr. Bender's practice has been largely in the areas of intellectual property litigation and counseling. He served as an Engineer with the Ford Motor Co., and as a Mathematician with Hughes Aircraft Co. He is the author of the treatise COMPUTER LAW: SOFTWARE PROTECTION AND LITIGATION (1984), and of numerous law review and continuing legal education articles. He is a frequent speaker on continuing legal education programs in the areas of patent-antitrust and computer law.

This article was written while the author was General Attorney, Intellectual Property Litigation, AT&T Technologies.
I. INTRODUCTION

In the nascent spectrum of legal expertise known as “computer law,” few segments have grown more rapidly over the past decade than software protection. And in the realm of intellectual property law, no subject matter has generated more interest during that period than software. Indeed, at the intersection of these two areas of law,
interest has mushroomed, as evidenced by a sharp increase in demand for counseling on how best to protect software, and by the crescendo of litigation directed toward its protection.

Before one can make knowledgeable choices among the various mechanisms and techniques available for protecting software, it is desirable to inquire into the inherent nature of software and its development, its traditional sources, and the legal and extra-legal methods available for its protection. Perhaps the best place to begin is by defining the subject matter.

"Software" is a term used somewhat differently by various authors, but which generally comprises three classes of subject matter: computer programs ("programs"); data bases; and documentation. For present purposes, a program can be regarded as a series of instructions in machine-readable form, prepared to achieve a certain result. A program, therefore, is the intelligence communicated to the computer by the human in the latter's attempt to get the machine to do his or her bidding. A data base is a machine-readable representation of information. It may be an employer's list of employees arranged alphabetically, a firm's list of customers arranged by zip code, etc. It is entered into the computer, which then operates upon it. Documentation refers to all documents (generally human-readable) which explain the operation of other software or of computer hardware. For example, documentation might discuss the methods used by a particular program to achieve its result, or might describe how to use the program.

This article shall concentrate on the program, although parts will apply to data bases. Documentation is generally considered ancillary to some other entity (e.g., a program) and is rarely treated separately.

II. UNUSUAL ASPECTS OF THE COMPUTER PROGRAM

What is so "special," technically, commercially, or legally, about programs that they merit such attention?

In some respects program protection may be treated as an instance of process protection. In other respects, however, the program is unusual, if not unique. One important reason why programs may be "special" is their ubiquity. By the end of 1982, there were more computers than people. Lowel, Digits to the
system to the carburetors of our automobiles. Even when we travel into space we cannot escape them — indeed, we cannot so travel without them.

Technologically, the program is quite unusual in that it may assume a number of different forms, which makes it confusing to understand. Commercially, programs are unusual in that some people, even those technologically "savvy" in other areas, often do not appreciate the value of programs. In the early days of the computer industry vast numbers of valuable programs were made available to users, free of charge. While few still give their programs away, many still do not fully realize the value of programs. Perhaps this failure to recognize, or accord full value to, this distinct corporate asset is attributable to the mystique which sometimes surrounds the computer. If so, it will doubtless dissipate along with that mystique.

Another reason why the program is arguably "special" is that it has already achieved enormous value, and is probably by far the fastest growing major type of technical subject matter. Total investment in software in the U.S. exceeds $200 billion, and the industry is in its infancy.

Legally, programs are quite unusual because they are surrounded by uncertainty with regard to each branch of law which may arguably be used to protect them.

III. THE NATURE OF A COMPUTER PROGRAM

A program can exist in a number of different forms, and recognition of this fact is necessary to effect any successful protection effort.


In 1974, less than 10% of our data processing costs went to software. Today, the figure is 50%, and by 1990 it is predicted to rise to 80%. McClellan, End of the Hardware Era, Datamation, May 1, 1984, at 122.

In 1981, more than $8 billion in software was marketed. Adapso, Sixteenth Annual Survey of the Computer Industry 53 (1982). In 1983, $2.1 billion in program products were marketed, with an estimate of $12 billion by 1988. Parcelling out the Software, N.Y. Times, Dec. 14, 1983, at D1, col.2. Those figures do not include the gigantic sums of money spent by companies to create and maintain software for their own use.

The program germinates in the minds of the persons who desire and/or create it. The first precursor of a program is a vision of some vague creature needed to fulfill a specific function. This is the "requirements" or "problem definition" or "specification" stage.

In the minds and hands of its designers, it takes a more definite form. The entity is broken into parts (modules), each of which is further subdivided. The nature, relationship, and interfaces of these parts are determined: This is the "architecture" stage.

The specific methods for solving the problem are fashioned. This is the "logic" or "algorithm" stage. This may involve a "flowchart," wherein the program's logic is depicted graphically by a sequence of oddly shaped boxes connected by lines and arrows, meant to show the various processing steps and flow of data. The flowchart is effectively the program's blueprint.

The program is then coded (the "coding" stage), resulting in a sequence of instructions in a rigidly defined language which typically has a vague resemblance to simple English. This is the "source program" and, while comprehensible to the programmer, it is not directly intelligible to the computer. Accordingly, the source program is translated into an "object program," a sequence of "bits" (zeroes and ones) which the computer understands (but which is not easily intelligible to the programmer). As the program is coded, much time will be spent testing it, to exorcise as many of the inevitable programming errors as possible (de-bugging). Finally, the persons who have designed and created the program create the documentation describing it.

In either source or object form, the program can be represented on various media. It can be handwritten on coding sheets (generally not machine-readable), or embodied on cardboard cards in the form of small rectangular punched holes or on tape in the form of magnetic impulses, etc. The machine-readable embodiment of the program (the deck of punched cards) is a "programming device," and is sometimes loosely called "the program." But in a more fundamental sense, the program is the intangible set of instructions embodying the architecture, logic, and testing, rather than any tangible embodiment. The set of source program instructions can be printed out in sequence, to form the "listing."

---

5. An algorithm is a rigidly defined, step-by-step procedure for solving a particular problem in a finite number of steps. The algorithm yields a solution to the particular problem under all circumstances.
IV. SOURCES OF COMPUTER PROGRAMS

There are a number of sources for programs. Perhaps the most obvious is the hardware vendor, and most (if not all) computers are sold with some programs included in the price. The programs most likely to be included are those generic to the hardware's operation, regardless of the specific task required by the user; these are called "system" or "control" programs. Beyond that, most manufacturers also offer "application" programs, which are directed to a particular function such as printing out a payroll, or updating an accounts receivable file. Today the manufacturer typically includes some programs in the price of the hardware, and licenses others for an additional fee.

A second source of programs is the user's own company. The majority of programs in use are probably user-created. When an organization has a specific problem for solution by computer, its first impulse typically is to create the necessary program in-house. Billions of dollars worth of programs are so created annually; most are of immediate use only to the organization creating them.

A third way of obtaining a program is to commission a "software house" to create it. Most software houses will provide, under contract, "custom" software to the specifications of those desiring it. A fourth way to get programs is through user groups or software clearing houses.

A fifth, and increasingly important way of obtaining use of programs, is to secure rights in a "program package" (also called a pro-


7. Ten years ago, "almost all" programs were supplied by the hardware manufacturers (presumably this excludes user-created programs). Today, independent software houses account for one third of the $14 billion dollar annual market, predicted to rise to a half by 1988. McClellan, supra note 4, at 122.

Some estimate that IBM's 1980 revenues for separately licensed software exceeded $1 billion. Missing Computer Software, supra note 4, at 46. IBM's 1984 revenues for separately licensed software have been forecast to be 5 billion. Verity, Call Unbundling Sign of Future Software-Driven Industry, ELECTRONIC NEWS, Nov. 10, 1980, § II, at 13.

8. Software houses are companies whose business is largely the creation, maintenance, and modification of programs. One estimate places the number of software houses at 4,000 in 1984. McClellan, supra note 4, at 122. In 1980 there were over 8,000 software houses with over 30,000 customers. Rush of New Companies to Mass-Produce Software supra note 4, at 54. The "independent" (i.e., non-mainframe manufacturer) software companies had 1980 revenues of $2 billion, and could top $8 billion in 1985. No Patents — But Still Protected, Bus. Week, Oct. 20, 1980, at 102.

9. See Gordon & Starr, supra, note 6 at 493.
proprietary program, program product, or canned program). Unlike hardware, software reproduction involves no bending of metal, or other labor-, space-, and material-intensive activity. As a result, once the program has been created, even if its creation cost over a million dollars, creating an additional copy (e.g., a duplicate magnetic tape embodying the program) costs less than a hundred dollars. A developing company could hope to market a program of widespread utility to numerous users. Each user would pay far less than the cost of creating the program, while the creating company, by marketing to many users, could hope to recoup far more than its cost. Accordingly, everyone might come out ahead.10

Some believe the program product did not begin to blossom until 1969, when IBM announced it was “unbundling.”11 In 1970, program package revenues were $25 million.12 By 1972 there were some 60 program products each of which had achieved lifetime fees of a million dollars.13 By 1977 their number had grown to 302,14 and by mid-1980 their number increased to 432.15 Some programs have been marketed to over a thousand installations.16 By 1979, program products had annual revenues well in excess of a billion dollars, with a predicted annual growth rate through 1985 of over 20% (twice as high as the computer hardware business).17

The program package has several possible advantages vis-a-vis a custom program: it can be operational much sooner; it should be

---

10. A good program product is more than simply a program available for licensing. It must be extensively debugged, reasonably easy to use, and accompanied by adequate documentation. It must also have a reasonably long useful life (preferably at least five years), widespread utility, improvement potential, and the ability to compete well against alternative program products. Finally, it must be priced right.

11. “Bundling” was the practice whereby computer manufacturers sold their hardware and software together as a single unit. The user did not have to use the manufacturer's software, but he did have to pay for it. See McGee, Financial Tax Accounting for Computer Software, 7 W. NEW ENG. L. REV. 651, 652-53 (1985).

12. Rush of New Companies to Mass-Produce Software, supra note 4 at 54.


14. Software Winners' Ranks Swell, Computerworld, March 27, 1978, at 2. Two of these each account for over $50 million. Id.


much less expensive; and by eliminating much "re-inventing of the wheel," it makes a more economic use of what we are increasingly told is a scarce resource — programmers.\textsuperscript{18} Some possible disadvantages vis-a-vis a custom designed program are longer running time, increased memory requirements, and the necessity for expensive and time-consuming alterations to adapt it to the user's situation.

The manner in which the user generally acquires a right to use in a program package is by taking a license. The license may also be the mechanism for acquiring rights in custom-designed programs, but more frequently such programs will simply be sold, rather than licensed, to the user (such a sale, because of the attributes of programs, involves its own intricacies).\textsuperscript{19} Further, licensing is one mechanism by which users acquire rights in manufacturer-created software. As a result of all this, licensing is by far the most important vehicle for acquiring rights in software created by another.

V. LEGAL METHODS FOR PROTECTING SOFTWARE

As a form of intellectual property, software is arguably protectable under patent law, copyright law, and/or trade secret law. As to patents, uncertainty exists with respect to applicability to most programs, although it is now clear that certain programmable processes and programmed machines are appropriate for patent protection. As to copyrights, there appears to be extensive applicability, but uncertainty exists with respect to the scope of protection. And while classical trade secret law clearly applies to programs, there is a question as to whether that venerable corpus of law still exists in its traditional form.

A. Patent Protection of Programs

1. Nature of Patent Grant

A United States patent grants to its owner the right to exclude others from making, using, or selling the claimed invention in the United States for seventeen years.\textsuperscript{20} It does not, however, confer on

\textsuperscript{18} As of 1980, although there were some 543,000 programmers and systems analysts in the United States the average Fortune 1000 company was estimated to have two years of programming waiting to be done. Bulkeley, supra note 15, at 1.

\textsuperscript{19} See Gordon & Starr supra, note 6, at 493.

\textsuperscript{20} Patent protection is granted in 35 U.S.C. § 154 which states in pertinent part: Every patent shall contain a short title of the invention and a grant to the patentee, his heirs or assigns, for the term of seventeen years . . . the right to exclude others from making, using, or selling the invention throughout the United States.
the owner the exclusive right to practice that invention, because one or
more other patents may also cover the invention. So, for example, if a
patent for a carburetor co-existed with a patent for a dual-barrelled
carburetor, the owner of the latter could not lawfully manufacture his
invention without a license from the owner of the former. However,
seventeen years after a patent issues, the owner loses all right to
exclude.

2. Obtaining a U.S. Patent

One seeks a patent by filing and prosecuting a patent application
with the U.S. Patent and Trademark Office (PTO). The application
should "contain a written description of the invention, and the manner
and process of making and using it, in such full, clear, concise, and
exact terms as to enable any person skilled" in the appropriate tech­
nology to make and use it; a statement of the best mode contemplated
by the inventor; and one or more claims pointing out the subject mat­
ter regarded as the invention. These claims specifically delineate the
metes and bounds of the invention, much as a deed sets forth the
boundaries of real estate.

The preparation and prosecution of a patent application is a tech­
nical and detailed task. An application can be filed and prosecuted
only by the inventor or by a registered patent attorney or patent agent.
Each year roughly 100,000 patent applications are filed and some
70,000 patents issue. The average time between filing of the applica­
tion and issue of the patent is currently between 2 and 2 1/2 years.

3. Requirements for a U.S. Patent

There are many requirements which must be satisfied for a valid
U.S. patent to issue. First, there is a strict standard of novelty. Beyond
this, there is a requirement that the invention not be "obvious" to one skilled in the appropriate area of technology. While these

---

Id.

21. Id. § 112.
22. Id.
"'Every man, woman, and child is a potential inventor,' says Isaac Fleischman, longtime
public information officer for the U.S. Patent Office, 'and 90% of them have tried to invent
something. . . ." Id. "[O]ver 100,000 patent applications are filed each year, and in 1973
54,960 U.S. Citizens were granted patents, which is one patent per 3,790 persons." Id.
24. Id. at 83.
26. Id § 103.
requirements are generic and not limited to program-related inventions, they will preclude patents on most programs.

4. Statutory Subject Matter

a. What Types of Inventions can be Patented?

A patent may be obtained on any useful, new and non-obvious "process, machine, manufacture, composition of matter, or any new and useful improvement thereof."27 Most (but not all) of the litigation, as well as writing, on whether programs are protectable by patent has been on this "subject matter" issue. Specifically, the question posed has been whether or not the claimed invention constitutes a programmable "process" or programmed "machine," as those terms are used in patent law. Basically, this has been reduced to the question of whether the invention is a mathematical algorithm (non-statutory) on the one hand, or a non-mathematical algorithm, or the application of an algorithm (statutory), on the other.

b. U.S. Supreme Court Cases

The subject matter issue has been considered on four occasions by the United States Supreme Court. In *Gottschalk v. Benson*,28 the Court held claims non-statutory for wholly pre-empting an "algorithm."29 The question left by the ambiguous language of that decision was whether a claim was rendered non-statutory for preemption of any algorithm, or rather solely for preemption of a mathematical algorithm.

In *Parker v. Flook*,30 the court stated that any mathematical algorithm in the claim must be treated as well-known, and if no novelty remains, the claim is not patentable.31 The court also asserted that a claim directed to an improved method of calculation using a formula, even if the solution to a particular end use, is non-statutory.32 A vigorous three-judge dissent accused the majority of confusing the subject matter standard with the novelty standard.33

In *Diamond v. Diehr*,34 a five to four majority decided a number of important points. First, they held that in determining whether a

27. Id. § 101.
29. Id. at 71-72.
31. Id. at 592-94.
32. Id. at 594-95.
33. Id. at 600.
34. 450 U.S. 175 (1981).
claim is statutory, "claim dissection" into old and new portions is improper; the claim must be analyzed as a whole, rather than at the "point of novelty" as the PTO had been inclined to do. Accordingly, the Court divorced section 101 subject matter considerations from section 102 novelty considerations. In addition, the Court interpreted Benson's use of the term "algorithm." In Benson, the question left undecided by ambiguous language was whether the court's ruling was limited to mathematical algorithms or was meant to extend to all algorithms. Diehr provided a clear answer: The Benson rule was limited to mathematical algorithms. Finally, the court specifically held that the statutory nature of a claim "is not altered by the fact that in several steps of the process a mathematical equation and a programmed digital computer are used."

Diamond v. Bradley, which the Court affirmed four to four without opinion, also stands for the proposition that the mere fact that a computer is involved in the claim cannot negate the presence of statutory subject matter.

c. PTO Guidelines

Shortly after the Supreme Court decided Diehr and Bradley, the PTO issued guidelines. The test embodied in these guidelines is to first determine whether a mathematical algorithm is directly or indirectly recited in the claim, and if so, to ask if the claim merely recites a mathematical algorithm. If the answers to both questions are in the affirmative, the claim is non-statutory; otherwise it is statutory.

d. Lower Court Cases

In the wake of Diehr and Bradley, the CCPA decided several cases providing useful standards and applications. In In re Taner, the court held that a claim reciting the "summing" of electrical signals

35. Id. at 188-89.
37. Id. § 102.
38. 450 U.S. at 184-86.
39. Id. at 186, n.9.
40. Id. at 185.
42. Id.
44. Id.
45. Id.
— which the court characterized as a mathematical algorithm — was actually directed to a conversion of electrical signals, deemed to be a statutory process, even though the "physical apparitions" could be expressed in mathematical terms.47

In re Pardo48 reiterated the two-step test to be used in determining whether program-related claims are statutory: (1) to determine whether a mathematical algorithm is directly or indirectly recited; and (2) if it is, to determine from analyzing the claim as a whole, whether the algorithm is applied in any manner to physical elements or process steps. If the answer to both questions is affirmative, the claim is statutory.49 The court applied that test to the steps of examining, compiling, storing, and executing formulas internally in a computer so as to convert it from a sequential processor to one whose functioning is not dependent on the order in which it receives instructions. The court ruled there was no mathematical algorithm directly or indirectly recited in these steps.50

In In re Abele,51 the court dealt with an algorithm conceded to be mathematical. The court held that one claim presenting no more than calculation and subsequent display of a number was non-statutory.52 However, another claim which required the input of data resulting from passing an X-ray through a CAT scanner was held statutory.53 The production, detection, and display steps were statutory, regardless of the presence of an algorithm, and it was in connection with them that the algorithm was applied. Accordingly, the rule embodied in Abele seems to be that if a mathematical algorithm is applied in any manner to physical elements or process steps, provided that the application is circumscribed by more than a field of use limitation or non-essential post-solution activity, the claim is statutory.54

In re Meyer55 dealt with claims for analyzing the results of tests on a complex system. Although no mathematical formula was recited in the claims, the court held that the decisive factor was whether the method was essentially a mathematical calculation.56 The purpose of

47. Id. at 790-91, 214 U.S.P.Q. (BNA) at 681-82.
49. Id. at 916, 214 U.S.P.Q. (BNA) at 676.
50. Id. at 916-17, 214 U.S.P.Q. (BNA) at 677.
52. Id. at 908, 214 U.S.P.Q. (BNA) at 687-88.
53. Id.
54. Id.
56. Id. at 795-96, 215 U.S.P.Q. (BNA) at 198.
the invention was partially to replace, with a computer, the thought processes of a neurologist. The applicant acknowledged that the claims recited a mathematical algorithm. The court concluded the algorithm had not been applied to physical elements or process steps and was, therefore, non-statutory.\textsuperscript{57}

5. Infringement

In the very earliest days of the computer industry, there was no such thing as a program. Computers were instructed through numerous circuits made by connecting wires and jacks on large structures resembling telephone switchboards. This was extremely cumbersome and time-consuming, and was soon replaced by the stored computer program. Crucial to an understanding of programs, however, is the tenet that there is an equivalence between software (the program) and hardware (the wired circuit). For every program there exists an equivalent hardware circuit, and vice versa.

Moreover, there is an entity which stands conceptually between hardware and software; aptly enough, it is called "firmware" or "read-only memory" (ROM). ROM comprises part of the computer's memory, indelibly imprinted with a program. This is in contrast to the software situation, where programs are read into memory, which is then effectively erased seconds or minutes later, when the instant job is terminated and the computer is readied for the next. There is generally full functional equivalence among hardware, software, and firmware.

This equivalence raises some interesting questions. First, can use of a programmed computer infringe a hardware patent? One court has answered that question in the negative\textsuperscript{58} while another has ruled that a "means plus function" apparatus claim, where hardware is disclosed, is infringed by a programmed digital computer.\textsuperscript{59} Another interesting — and thus far unanswered — question is whether, where a "program patent" issues, equivalent hardware may infringe it.

\textsuperscript{57} Id. at 796, 215 U.S.P.Q. (BNA) at 198-99.


6. Advantages and Disadvantages of Patent Protection

The advantages of patent protection for software include the following: (1) patent protection is well suited to an invention which may enjoy great proliferation; (2) patents protect against independent invention (one corollary is that in litigation, the plaintiff need not show copying); (3) the protective duration is seventeen years, relatively long in the program milieu; (4) once secured, patent protection may be maintained with relative ease\(^{60}\) and aids in licensing; and (5) because the owner has an exclusive right in the algorithm he or she may freely disclose how the program works — which may be an important marketing tool.

Among the disadvantages of patent law in the software milieu are the following: (1) it is difficult to obtain preliminary relief in the event of litigation;\(^{61}\) (2) many types of programmable processes and programmed machines may ultimately be held to be non-statutory subject matter; (3) there is present commercial uncertainty arising from this fact, even if many types of programs should eventually be held patentable; (4) most valuable programs will not qualify for patents even if the subject matter is appropriate since they will run afoul of the novelty and non-obviousness requirements; (5) even if a program may be patented, this may be possible only if it is claimed in an awkward and unnatural manner; (6) the cost of securing patent protection may be relatively high;\(^{62}\) (7) the interim between filing a patent application and issue of a patent typically exceeds two years;\(^{63}\) often a significant part of the program’s lifetime; (8) patent protection is not applicable to databases or documentation; (9) because the patent itself is a public document, it is difficult to police unlawful use; (10) patents may have a more restricted ambit of proper exploitation than alternative methods;\(^{64}\) (11) if the owner is party to an antitrust consent judgment requiring him to license all patents, such patents are presumably included; and (12) aside from any problems unique to software, the


\(^{61}\) Some feel that the law may be about to change on this point. See Smith Int’l v. Hughes Tool Co., 718 F.2d 1573 (Fed. Cir.), cert. denied, 104 S. Ct. 493 (1983).

\(^{62}\) In the case of a patent which issued in 1980 on a sort process embodied in a popular program product, it was reported that the cost to prosecute the application was about $100,000. Rosenberg, Whitlow Sort Processes Patented, Computerworld, July 28, 1980, at 55.

\(^{63}\) Prosecution of the patent referenced in the immediately preceding note required some eight years. \textit{Id.}

\(^{64}\) The doctrine of patent misuse renders unenforceable a patent used in a manner deemed by a court to be contrary to public policy. Morton Salt Co. v. G.S. Suppieger Co., 314 U.S. 488, 52 U.S.P.Q. (BNA) 30 (1942).
entire U.S. patent system has problems manifested both in the PTO and the courts.65

B. Copyright Protection of Software

Copyright protects an author's original expression or form, as opposed to idea or concept. It protects against copying but not against independent origination. The problems inherent in the application of copyright law to programs include whether programs constitute copyrightable subject matter, and if so, determining the scope of copyright protection. The Copyright Office long ago announced a policy favoring copyrightability of programs.66

1. Statutes

The new Copyright Act, generally effective January 1, 1978,67 contains section 102(a):

Copyright protection subsists, in accordance with this title, in 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, whether directly or with the aid of a machine or device . . . . 68

The Act, as enacted in 1976, also contained section 117:

Notwithstanding the provisions of Sections 106 through 116 and 118, this title does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information, or in conjunction with any similar device, machine, or process, than those afforded to works under the law, whether title 17 or the common law or stat-

---

65. Positive steps taken to rectify this problem include (a) creation in 1982 of the U.S. Court of Appeals for the Federal Circuit, to which all patent infringement appeals now go. See 28 U.S.C. §§ 1292(c)-1292(d), 1295 (1982), and (b) increased funding for the PTO.


From 1976 through mid-1980, only 2,000 programs were registered with the Copyright office. No Patents—But Still Protected, Bus. Week, Oct. 20, 1980, at 102. However, this was not indicative of the degree of reliance on copyright, as registration, although necessary for enforcement, may not be necessary for protection. In the past few years the number of registrations for programs has increased sharply.


utes of a State, in effect on December 31, 1977, as held applicable and construed by a court in an action brought under this title. 69

A commission was created by Congress 70 to study the problem of protecting programs. CONTU recommended legislation:

The new copyright law should be amended: 1) to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright; 2) to apply to all computer uses of copyrighted programs by the deletion of present Section 117; and 3) to ensure that rightful possessors of copies of computer programs may use or adapt these copies for their use. 71

The Computer Software Copyright Act of 1980, embodying the CONTU recommendations, was enacted into law in December 1980. 72

It added a definition to section 101 of the Copyright Act:

A "computer program" is a set of statements or instructions to be used directly or indirectly in a computer in order to bring about a certain result. 73

It also replaced the original section 117:

§ 117. Limitations on exclusive rights: computer programs.

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

(1) that such a new copy or adaptation is created as an essen-

---

69. Id. § 117.
71. FINAL REPORT OF THE NATIONAL COMMISSION ON NEW TECHNOLOGICAL USES OF COPYRIGHTED WORKS, 1 (1978) (hereinafter CONTU REPORT). Commissioner John Hersey, the noted author of the CONTU REPORT, dissented vigorously on the basis of his conviction that a computer program is not a "writing" in the Constitutional sense: "The Congress shall have the power . . . To promote the Progress of Science [i.e., philosophy] . . . by securing for limited Times to authors . . . the exclusive Right to their Writings. . . ." U.S. CONST. art. 1, § 8. "The Act of 1976 should be amended to make it explicit that copyright protection does not extend to a computer program in the form in which it is capable of being used to control computer operations." CONTU REPORT at 37.
(2) that such a new copy or adaptation is for archival purposes only and that all archival copies are destroyed in the event that continued possession of the computer program should cease to be rightful.

Any exact copies prepared in accordance with the provisions of this section may be leased, sold, or otherwise transferred, along with the copy from which such copies were prepared, only as part of the lease, sale, or other transfer of all rights in the program. Adaptations so prepared may be transferred only with the authorization of the copyright owner.

One possible problem posed by the statute is whether it is Constitutional. Some have suggested that defining a program as copyrightable subject matter may run afoul of the Constitutional dichotomy between the domains of the patent (excluding practice of the tangible embodiment of an idea) and of the copyright (protecting the copying of an expression). This argument traces its heritage back at least as far as Baker v. Selden, which held that where the use of copyrighted forms was necessary to practice an accounting method explained in a book containing the forms, it was not copyright infringement to so use the forms. Further, some read Selden to hold that if it is impossible to separate idea from expression, use of the expression cannot constitute copyright infringement. The patent/copyright relationship was further explored in Mazer v. Stein, wherein the court held that a particular statuette lamp base was copyrightable subject matter because its aesthetic qualities were separable from its utilitarian aspects.

More recently, the second circuit decided whether certain three dimensional artistic belt buckles were copyrightable under the old Act.

74. Indeed, 17 U.S.C. § 102(b) (1982) provides:
To 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.
75. See Commissioner John Hersey's dissent in the CONTU REPORT, supra note 71, at 27-37.
76. 101 U.S. 99 (1879).
77. Id. at 101-104.
and the new. The trial court held that they were not because they “fail[ed] to satisfy the test of separability and independent existence of the artistic features, which is required under both statutes.” On appeal, the defendant argued the buckles were uncopyrightable as being “useful articles” without sculptural features that could be identified separately from, or were capable of existing independently of, the utilitarian aspects. Noting that separability may occur either physically or conceptually, the second circuit held the buckles copyrightable because the “primary ornamental aspect . . . is conceptually separable from the subsidiary utilitarian function.”

Judge Weinstein dissented, asserting that the innovations of artistic form were inseparable from the function and that the 1976 Act protects only those useful articles which can be identified separately from, and are capable of existing independently of, utilitarian aspects. He noted further that the “overall design or configuration of a utilitarian object, even if it is determined by aesthetic as well as functional considerations, is not eligible for copyright.” According to the dissent, the distinction is between “ornamental and superfluous designs contained within useful objects” and “artistically designed functional components of useful objects.” Indeed, Judge Weinstein stated that in propounding the 1976 Act, Congress considered and rejected extending copyright protection to the design of a useful arti-

82. Id. Both the old and new law deny copyrightability to such an article. See 17 U.S.C. § 101 (1982).
83. Kieselstein-Cord v. Accessories by Pearl, Inc., 632 F.2d 989, 993, 208 U.S.P.Q. (BNA) 1, 5 (2d Cir. 1980). No claim was made that the buckles lacked originality or creativity, and the court stated that they “[rose] to the level of creative art.” Id. at 994, 208 U.S.P.Q. (BNA) at 2.
84. Id. at 994, 208 U.S.P.Q. (BNA) at 6 (Weinstein, J., dissenting). “It is the originator’s success in completely integrating the artistic designs and the functional aspects of the buckles that preclude copyright.” Id.
85. Id. at 995, 208 U.S.P.Q. (BNA) at 6 (Weinstein, J., dissenting).
86. Id. at 996, 208 U.S.P.Q. (BNA) at 7 (Weinstein, J., dissenting) (quoting Esquire, Inc. v. Ringer, 591 F.2d 796 (D.C. Cir.), cert. denied, 440 U.S. 908, reh’g denied, 441 U.S. 917 (1979)).

Indeed, one author suggests that as applied to artistic writings the idea/expression “dichotomy is at best difficult to apply and at worst wholly inapplicable. . . . copyright does protect some ideas.” Note, Derivative Works and the Protection of Ideas, 14 GA. L. REV. 794, 812 (1980).
Acknowledging the policy question of whether encouraging art is more important than permitting the less affluent to enjoy such works, he left this question for Congress.89

2. Copyright Cases Relating to Software Protection

Although the first such case was not decided until 1978, cases are now cascading through the lower courts. These cases suggest there may be no area in which it is more important for courts to grasp the fundamentals of computer technology than in the area of copyright protection for software. The first case dealing squarely with infringement of a copyright on a program is *Data Cash Sys., Inc. v. JS&A Group.*90 Defendant marketed a game including a ROM which embodied plaintiff's program. Plaintiff had no copyright notice, except on all copies of the source program (used internally by plaintiff), for which copyright registration was secured a year after plaintiff began marketing the program.91

Plaintiff sued for copyright infringement and sought a preliminary injunction. Defendant moved for summary judgment.92 In both ROMs, the program was in object form.93 The district court concluded that "the 'source program' is a writing while the 'object program' is a mechanical tool or machine part."94 The court determined that the applicable law was the old copyright law so that either the common law or the 1909 Act would control on the issue of whether a ROM was a copy.95 Further, the court held that under either standard the ROM was not a copy. "In its object phase, the ROM, the computer program is mechanical tool or a machine part but it is not a 'copyright' of the source program."96

Finally, the court noted in dictum that if the 1976 Act did apply, copying a ROM would not constitute infringement because, while a ROM is a "copy" under that Act, that Act did not apply to programs

88. *Id.* at 997, 208 U.S.P.Q. (BNA) at 8 (Weinstien, J., dissenting).
89. *Id.* at 999, 208 U.S.P.Q. (BNA) at 9-10 (Weinstein, J., dissenting). In 1981 legislation was introduced in the House to amend the Copyright Act so as to "provide for protection of ornamental designs of useful articles." H.R. 20, 97th Cong., 1st Sess.
92. *Id.* at 1065, 203 U.S.P.Q. (BNA) at 738.
93. *Id.* at 1066, 203 U.S.P.Q. (BNA) at 738.
94. *Id.* at 1065, 203 U.S.P.Q. (BNA) at 738.
95. *Id.* at 1067, 203 U.S.P.Q. (BNA) at 740.
96. *Id.* at 1066-67, 203 U.S.P.Q. (BNA) at 741.
"in their object phase, i.e., ROM." 

The court of appeals affirmed summary judgment, stating it was unnecessary to reach the issue of whether a ROM was a copy, and ruling that when it entered the public domain, the work lacked proper copyright notice so that under the applicable 1909 Act, copyright was forfeited. While purporting not to consider whether under the 1909 Act a ROM constituted a copy of the program embodied in it, the court — apparently without considering that issue and in the face of its statement that it need not reach it — implicitly decided in the affirmative. For in determining that publication occurred prior to 1978, the court looked to section 26 of the 1909 Act, which defined "date of publication" as the earliest date when "copies" were placed on sale.

A somewhat earlier case, also of interest for its reasoning, is Synercom Technology, Inc. v. University Computing Co., which raised the question of whether a program could infringe a copyright on an input format form. Defendant developed a program similar in function to plaintiff's, but which accepted its input in a different arrangement. Because plaintiff had a significant market position, defendant determined that defendant's marketing potential would increase if the input prepared by plaintiff's users for use with plaintiff's programs, could also be used with defendant's programs.

Accordingly, defendant created a "preprocessor program" whose function was to take input designed for plaintiff's program, and to con-

97. Id. at 1066-67 n.4, 203 U.S.P.Q. (BNA) at 739-740 n.4. Two reasons were stated: (1) "In its object phase, the computer program is a mechanical device which is engaged in the computer to become an essential part of the mechanical process," and "[m]echanical devices which cannot qualify as pictorial [or] graphic ... works are not writings and may not obtain copyright protection;" and (2) the Copyright Office's then proposed (and subsequently adopted) regulations on affixing notice contained language which seemed not to contemplate copyrightability of ROM's. Id.

98. Id. at 1041, 208 U.S.P.Q. (BNA) at 199.


100. Id. at 1042, 208 U.S.P.Q. (BNA) at 200.


102. An "input format" is the arrangement of data to be input to the computer, so that the computer will recognize each item for what it is and for what its form is. As to each item there is a prescribed location (e.g., sixth location in the format), unit (e.g., meters), and a particular manner of presentation (e.g., six digit decimal form with two digits to the right of the decimal point). Format is a concept that may be expressed by means of input format forms, which are specially prepared forms permitting insertion of input data in the proper arrangement and form.

vert it into the form of input for defendant's program. Plaintiff contended that defendant's preprocessor infringed the copyright on plaintiff's format form. "The [programming] statements in its preprocessor program are derived directly and precisely from the copyrighted manual card formats." Looking to section 101 of the 1909 Act, which granted to the copyright owner the exclusive right "to translate the copyrighted work into other languages or dialects, or make any other version thereof," plaintiff contended that the preprocessor infringed because it did "no more than translate the expression of the formats to a different computer language."

The court wrestled mightily with the question, asking if "sequencing and ordering is expression, what separable idea is expressed?" The court concluded that the formats would be copyrightable if the ideas they expressed were separable from their expression; it ruled that they were not separable.

Plaintiff also apparently contended that defendant's preprocessor infringed the copyright in plaintiff's manual. The court stated that translating the copyrighted program from one computer language to another would infringe, and that programming a copyrighted flowchart would probably infringe. But preparing a program from a general description of the problem would be quite dissimilar to translation of a literary work and would not infringe:

In most cases, the formulation of the problem in sufficient detail and with sufficient precision to enable it to be converted into an unambiguous set of computer instructions requires substantial imagination, creativity, independent thought, and exercise of discretion, and the resulting program can in no way be said to be merely a copy or version of the problem statement.

The most significant copyright software protection case to date may be Apple Computer, Inc. v. Franklin Computer Corp., which

---

104. Id. at 1006-07, 199 U.S.P.Q. (BNA) at 540-41.
105. Id. at 1012, 199 U.S.P.Q. (BNA) at 545.
107. Id.
109. Id. at 1013, 199 U.S.P.Q. (BNA) at 546.
110. Id.
111. Id. at 1004, 199 U.S.P.Q. (BNA) at 538.
112. Id. at 1013 n.5, 199 U.S.P.Q. (BNA) at 546 n.5.
inquired into the patent/copyright dichotomy. Plaintiff alleged copyright infringement as to system programs embodied in ROMs and floppy disks. In denying plaintiff's motion for a preliminary injunction, the district court was of the view that object code embodied in a silicon chip ROM might not be protectable by copyright as not constituting a fixed expression of an original work.

The third circuit reversed. Defendant did not dispute copying, and its factual defense was that it was not feasible for it to write its own operating system programs. Defendant's principal legal defense was that system programs were not susceptible to copyright protection.

The appellate court saw three issues: (1) whether copyright can exist in a program expressed in object code; (2) whether copyright can exist in a program embedded in ROM; and (3) whether copyright can exist in a system program.

On the first point, the third circuit found no statutory basis for distinguishing source code from object. It also found that its own opinion in Williams Electronics, Inc. v. Artic International, Inc., dealt with many of the doubts expressed here by the district court. The 1976 Act states two requirements for a work to constitute copyrightable subject matter: It must be (1) an "original work of authorship," and (2) "fixed in a tangible medium of expression." Section 102(a) enumerates seven types of "works of authorship," and the court found that the legislative history suggested that programs

---

114. Presumably code embodied in a floppy disc would be similarly unprotectable.
117. Id. at 1245, 219 U.S.P.Q. (BNA) at 117. Defendant seemed to be saying it was not feasible for defendant to rewrite the system software so as to be compatible with plaintiff's application programs. The espoused reason was that there were too many "entry points" in relationship to the number of instructions in the program. Entry points are those locations in the system program which can be used to mesh the application programs with the system program. Defendant also alleged that "identical signals" were necessary to ensure 100% compatibility. The evidence was mixed on whether in fact the system programs could have been rewritten so as to be compatible. Id.
118. Id.
119. Id. at 1246, 219 U.S.P.Q. (BNA) at 118.
120. Id. at 1247, 219 U.S.P.Q. (BNA) at 119.
121. 685 F.2d 870, 215 U.S.P.Q. (BNA) 405 (3d Cir. 1982).
124. Id.
were within the section 102(a) definition of literary works.\textsuperscript{125}

The CONTU Report recommended that the law be amended to make it explicit that programs, to the extent they embody an author's original creation, are copyrightable.\textsuperscript{126} The 1976 and 1980 versions of section 117 dealt only with the scope of protection. "The language of the [1980] provision, however, by carving out an exception to the normal proscriptions against copying, clearly indicates that programs are copyrightable and are otherwise afforded copyright protection."\textsuperscript{127}

The third circuit noted that in \textit{Williams Electronics}, it concluded that copyrightability of programs was established after the 1980 amendment, and it had there rejected the argument there was a distinction between copyrightability of a source code program and of its object code form.\textsuperscript{128}

The district court's suggestion that copyrightability depended on a communicative function to individuals emanated from the \textit{White-Smith Music Co. v. Appollo Co.},\textsuperscript{129} which held that a player piano roll was not a copy of a musical composition because it was not in a form people generally could perceive. The court suggested that the piano roll was not a copy because it was not intelligible to humans, but rather could be read only by a machine — a player piano.\textsuperscript{130} But the appellate court found it clear from the language and the legislative history of the 1976 Act that it was intended to obliterate the \textit{White-Smith} distinction.\textsuperscript{131} Moreover, the CONTU majority took the position that object code was appropriate for copyright;\textsuperscript{132} this, in the face of a dissent based on the theory that the machine control phase of a program is not directed to humans.\textsuperscript{133}

Turning to the copyrightability of a program embedded in a ROM, the court found that issue also decided in \textit{Williams Electronics}.\textsuperscript{134} Defendant there argued that copyright would not apply to a ROM because it was a utilitarian object or machine part. The \textit{Williams} court held that "fixation" was satisfied through embodiment of

\textsuperscript{125} Id. 685 F.2d at 875, 215 U.S.P.Q. (BNA) at 409.
\textsuperscript{126} CONTU REPORT, supra note 71, at 37.
\textsuperscript{128} Williams Elec., 685 F.2d at 876-77, 215 U.S.P.Q. (BNA) at 409.
\textsuperscript{129} 209 U.S. 1 (1908).
\textsuperscript{130} Id. at 18.
\textsuperscript{131} Williams Elec., 685 F.2d at 877 n.8, 215 U.S.P.Q. (BNA) at 410 n.8.
\textsuperscript{132} CONTU REPORT, supra note 71, at 38.
\textsuperscript{133} Id.
the expression in the ROM. Accordingly, the court reaffirmed that “a computer program in object code embedded in a ROM chip is an appropriate subject of copyright.”

The court next turned to what it viewed as defendant’s main contention: that system programs are excluded from copyright protection under 17 U.S.C. section 102(b), and under Selden. Defendant made two arguments: (1) an operating system program is a process, system, or method of operation and, therefore, uncopyrightable; and (2) in the case of system software, idea merges with expression so that the former cannot be used without reproducing the latter; accordingly, copyright cannot bar such use.

The court agreed with the defendant’s contention that underlying section 102(b) and Selden is the distinction between property subject to the patent law, and property subject to the copyright law. But the court believed that defendant misapplied the distinction. In the court’s view, the plaintiff sought to copyright not the method which instructs the computer to perform its operating functions, but rather the instructions themselves. “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 that it would be if instructions were written in ordinary English in a manual. . . .” Moreover, the court saw defendant’s contention as inconsistent with its concession that application programs are appropriate for copyright protection.

---

137. *Id.* Defendant viewed Selden as standing for three propositions: (1) use of a system does not infringe a copyright on the description of the system; (2) copyright does not extend to purely utilitarian works; (3) copyright may not be used to monopolize an idea. *Id.* at 1250, 219 U.S.P.Q. (BNA) at 121-22.
138. *Id.* at 1250, 219 U.S.P.Q. (BNA) at 122.
139. *Id.* at 1253, 219 U.S.P.Q. (BNA) at 124.
140. For the text of § 102(b) see *supra* note 74.
142. *Id.* at 1250, 219 U.S.P.Q. (BNA) at 122.
143. *Id.* at 1251, 219 U.S.P.Q. (BNA) at 123.

Both types of programs instruct the computer to do something. Therefore, it should make no difference for purposes of section 102(b) whether these instructions tell the computer to help prepare an income tax return (the task of an application program) or to translate a high level language program from source code into its binary language object code from (the task of an operating system program). . . .
The court concluded the defendant's argument that:

an operating system program is part of a machine mistakenly focuses on the physical characteristics of the instructions. But the medium is not the message. . . ." The mere fact that the operating system program may be etched on a ROM does not make the program either a machine, part of a machine, or its equivalent.\textsuperscript{144}

The court agreed with the CONTU majority that programs should no more be considered machine parts than videotapes should be considered parts of projectors.\textsuperscript{145}

The defendant also contended operating systems could not be copyrighted because they are purely utilitarian, and the plaintiff sought to block use of the art embodied in its operating system.\textsuperscript{146} The court saw this argument as stemming from dictum in Selden:

\ldots where the art [taught by a book] 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, given therewith to the public; not given for the purpose of publication in other works explanatory of the art, but for the purpose of practical application.\textsuperscript{147}

The court rejected what it saw as the expansive reading given this language by some other courts.\textsuperscript{148} Defendant's interpretation of this language precluded copyrightability where the work is put to use. However, the court saw that interpretation as rejected in Mazer v. Stein:\textsuperscript{149} "We see nothing in the copyright statute to support the argument that the intended use or use in industry of an article eligible for copyright bars or invalidates its registration."\textsuperscript{150}

\begin{footnotes}
\item[144] Id.
\item[145] Id. (quoting CONTU REPORT, \textit{supra} note 71, at 21).
\item[146] Id.
\item[147] \textit{Selden}, 101 U.S. at 103.
\item[149] 347 U.S. 201 (1954).
\item[150] Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1252, 219 U.S.P.Q. (BNA) 120, 123 (3d Cir. 1983) (quoting \textit{Mazer}, 347 U.S. at 218). The Third Circuit noted that the CONTU majority agreed "that the words of a program are used ultimately in the implementation of a process should in no way affect their copyrightability." \textit{Id.} (quoting \textit{CONTU REPORT, supra} note 71 at 21). "The copyright status of the written rules for a game or a system for the operation of a machine is unaffected by the fact that those rules direct the actions of those who play the game or carry out the process." \textit{Id.} The court considered "the CONTU Report as accepted by Congress since Congress wrote into the law the majority's recommendations almost verbatim." \textit{Id.}
\end{footnotes}
The court noted that under Selden, as interpreted in Mazer, "a copyright gives no exclusive right to the art disclosed; protection is given only to the expression of the idea — not the idea itself." The dichotomy is embodied in section 102(b), which the legislative history indicates was intended "to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law."

Noting that courts have found difficulty in drawing the line between idea and expression, the court stated that in the case of an operating system, the line must be a pragmatic one balancing competition on the one hand, against protection on the other.

If other programs can be written or created which perform the same function as an Apple's operating system program, then that program is an expression of the idea and hence copyrightable. In essence, this inquiry is no different than that made to determine whether the expression and idea have merged, which has been stated to have occurred where there are no or few other ways of expressing a particular idea.

Defendant claimed that whether or not the programs could be rewritten, there were a limited number of ways to arrange an operating system to enable a computer to run Apple-compatible application programs. According to the court this claim had no pertinence to the practice/expression dichotomy or to merger.

The idea which may merge with the expression, thus making the copyright unavailable, is the idea which is the subject of the expression. If other methods of expressing that idea are not foreclosed as a practical matter, then there is no merger. [Defendant] 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 expression, have merged.

Another basis for the court's conclusion was that the statutory

---

151. Id. (quoting Mazer, 347 U.S. at 217).
154. Id.
155. Id.
156. Id.
definition of a computer program did not distinguish between application and operating programs. The court noted that the district court in Apple Computer, Inc. v. Formula International, Inc. found no distinction, and that Tandy Corp. v. Personal Micro Computers, Inc. had upheld the copyrightability of an operating system without discussing the issue.

Accordingly, the denial of the preliminary injunction was reversed and the case was remanded.

The past four years have seen a cascade of cases involving program copyright, with the vast majority espousing the propriety of protecting object code in ROM form.

161. Id. at 1255, 219 U.S.P.Q. (BNA) at 126. The appellate court also reversed the district court finding that plaintiff had not made the requisite showing for irreparable harm, because the lower court failed to recognize that "a copyright plaintiff who makes out a prima facie case of infringement is entitled to a preliminary injunction without a detailed showing of irreparable harm." Id. at 1254, 219 U.S.P.Q. (BNA) at 125. Aside from that, the appellate court found substantial evidence of the considerable effort and investment by plaintiff in developing the programs in suit. Thus, even without the presumption of irreparable harm, plaintiff showed irreparable harm. Id.
162. Id. at 1255, 219 U.S.P.Q. (BNA) at 126. Because the record was unclear on whether some or all of the operating programs represented the only means of expressing the idea underlying them, the court stated that necessary findings could be made on remand. Id. at 1253, 219 U.S.P.Q. (BNA) at 124.
3. The Work Made for Hire Doctrine

Section 210(b) of the Copyright Act provides:

(b) Works made for hire — In the case of a work made for hire, the employer or other person for whom the work was prepared is considered to be author for the purposes of this title and, unless the parties have expressly agreed otherwise in a written instrument signed by them, owns all of the rights comprised in the copyright.\textsuperscript{164}

Section 101 defines a “work made for hire” as:

(1) a work prepared by an employee within the scope of his or her employment; or (2) a work specially ordered or commissioned for use as a contribution to a collective work, as a part of a motion picture or other audiovisual work, as a translation, . . . , if the parties expressly agree in a written instrument signed by them that the work shall be considered a work made for hire. . . .\textsuperscript{165}

To date there appears to be only one software-related case on this point. In \textit{BPI Sys., Inc. v. Leith},\textsuperscript{166} the plaintiff was developing an accounts receivable system and contracted with defendant to produce some twelve programs to be embodied in that system. The controlling issue was whether these programs constituted work made for hire, or in other words, whether the defendant was the plaintiff’s employee. The court decided the question in the negative,\textsuperscript{167} and denied relief.

An interesting case not related to software was \textit{Black},\textsuperscript{168} in which the court held that when an employer materially breaches the employment agreement, there arises a claim for rescission of the contract,

\begin{itemize}
  \item 164. 17 U.S.C. § 201(b) (1982).
  \item 165. Id. § 101.
  \item 167. \textsc{id}. This work was performed at [d]efendant’s place of business on his own computers. He was paid per program and no social security or income tax was deducted from his pay. Although [p]laintiff could accept or reject the final product, it had no right to control the manner of the work performed by [d]efendant. To aid him in the project, [p]laintiff supplied him with certain internal, confidential documents which were not available to the public . . . . \textsc{defendant} was not an employee of [p]laintiff acting within the scope of his employment. The work he did was specially ordered and commissioned but there was no written agreement that the work was ‘work made for hire.’
\end{itemize}
"including the implied right to authorship." By analogy, similar argument may be made in the absence of an employer/employee relationship where the parties have executed a writing specifying that the work is made for hire.

It is anticipated that the work made for hire doctrine may result in more problems in the program scenario than in more traditional areas of copyright protection. The reason is that in certain segments of the software industry it is common to take a working module which performs a particular function, and incorporate it in a second program where, with or without modification, it will perform a similar function. It may then be taken from the second program and, with or without modification, installed in a third. Further, loose consulting arrangements without written agreement often are the modus operandi of the software industry. Accordingly, this may give rise to more work made for hire problems than in an area such as book publishing.

4. Advantages and Disadvantages of Copyright Protection

Among the advantages of copyright protection for programs are: (1) the relative ease and inexpensiveness of securing and maintaining it; (2) its suitability for great proliferation; (3) the extraordinary duration of protection; and (4) the realistic possibility of securing preliminary relief in the event of litigation. Among the disadvantages are: (1) the uncertainty of its applicability to programs; (2) even more fundamentally, the uncertainty as to scope of protection; and (3) the difficulty of policing.

C. Trade Secret Protection of Software

1. Classical Trade Secret Protection

Trade secret is in fact the method which has been used more frequently than any other to protect software. A trade secret is "any formula, pattern, device or compilation of information which is used in one's business and which gives him an opportunity to obtain an advantage over competitors who do not know or use it." Basically, the traditional trade secret doctrine states that when a business has secret information which gives it a commercial advantage over com-

169. Id.
170. Indeed, under the 1976 Act, the program is copyrighted upon being represented in tangible form. 17 U.S.C. § 102 (1982).
172. RESTATEMENT OF TORTS, §757 comment b (1939).
petitors not in possession of the information, a court will afford protection against unauthorized use or disclosure, but not against one who independently develops. Programs, data bases, and documentation may all be protectable as trade secrets. Various characteristics and representations of programs such as algorithms, architecture, and flow charts may likewise be protectable.

Unlike its statutory cousins, the scope of whose applicability is in issue, traditional trade secret law clearly applies to programs. The scope of protection seems reasonably clear. Trade secret rights arise by operation of law in many situations where a confidential relationship is to be implied, such as the employer/employee relationship. But even in such situations, a well drafted contract may be preferred, as it can serve to make rights more specific, to give notice, and to create additional rights. A study conducted a year before enactment of the Computer Software Copyright Act of 1980 concluded that software developers would continue to rely most heavily on trade secret protection for their programs.

One interesting point is that trade secret protection is available against appropriation by memory; no tangible item need be carried off. This view was affirmed in a case dealing with software.

---

173. See generally Restatement of Torts §757, 758 (1939).
175. However, there is a tendency by courts to strictly construe such a contract, thereby extinguishing any pre-existing rights. Accordingly, care should be exercised in drafting the agreement. See Gordon & Starr supra, note 6 at 488-89.
177. Were such protection unavailable, trade secret protection of programs might be partly academic. Competent programmers and systems analysts spending much of their time with a program remember the program's structure, large segments of the algorithms and portions of the coding.
178. Support for this position is stronger for the technological trade secret than for the collection of information (i.e., the customer list, and presumably the data base).
other interesting point is that trade secret law has its criminal side.\textsuperscript{180} Aside from general larceny statutes which may apply,\textsuperscript{181} about half of the states have specific trade secret theft statutes.\textsuperscript{182} While United States copyright law has criminal sanctions (for willful infringement),\textsuperscript{183} the patent law has none.

2. The Preemption Problem

However, a recent challenge to classical trade secret protection is posed by the possibility of preemption for this entire corpus of law, or particular aspects of it, by Section 301(a) of the new Copyright Act:

On or after January 1, 1978, all legal or equitable rights that are equivalent to any of the exclusive rights that are within the general scope of copyright as specified by Section 106 in works of authorship that are fixed in a tangible medium of expression and come within the subject matter of copyright as specified by Sections 102 and 103, whether created before or after that date and whether published or unpublished, are governed exclusively by this title. Thereafter, no person is entitled to any such right or equivalent right on any such work under the common law or statutes of any State.\textsuperscript{184}

The existence of this section poses the issue of whether it preempts trade secret protection, wholly or in part, and the past few years have witnessed a number of cases dealing with possible preemption of diverse areas of law.

\textsuperscript{180} For a discussion of this aspect, see Bender, \textit{Trade Secret Protection of Software}, supra note 171, at 942-56. See also Bender, \textit{Trade Secret Software Protection}, supra note 171, at 59-65.

\textsuperscript{181} \textit{See} Hancock v. State, 402 S.W.2d 906 (Tex. Crim. App. 1966), aff'd sub. nom. Hancock v. Decker, 379 F.2d 552 (5th Cir. 1967) (per curiam). Indeed, the trade secret criminal statutes seem to be used more in the case of software than in the case of any other type of subject matter. Three reasons are often given as justification for the necessity of a criminal remedy: (1) difficulty of civil suit; (2) judgment-proof misappropriators; and (3) empty threat of civil suit where only nominal investment is necessary for misappropriator to exploit. The second and third bases are more likely to apply to a misappropriator of software than of most other types of subject matter.


\textsuperscript{183} \textit{17} U.S.C. § 506 (1982).

\textsuperscript{184} \textit{Id.} § 301(a).
In the trade secret area, one case held that a complaint alleging that defendant "without proper authority obtained and improperly utilized copies of . . . proprietary drawings" states a claim which is preempted by section 301(a) because "The facts relate to rights that are equivalent to the exclusive rights belonging to a copyright holder. . . ." The court believed preemption mandated where a claim does not allege the elements of "invasion of privacy, a trespass, a breach of trust or a breach of confidentiality," and the court ruled that this claim (specifically denominated as being for "misappropriation of trade secrets") failed to allege such an element. And in a case involving a computer program, a court held at least certain aspects of state trade secret law were pre-empted by federal copyright law.

In another case involving a computer program, however, another district court held that section 301 does not preempt, because a trade secret claim is not "equivalent" to copyright claim. "It is well settled that copyright protection extends not to an idea itself, but rather to the particular expression . . . . In contrast, the protection provided by the common law of trade secret misappropriation extends to the very ideas . . . ."

3. Combined Trade Secret and Copyright Protection of Programs

Some have suggested attempting to secure the advantages of both copyright and trade secret protection. In such a situation the owner marks the program with both copyright notice and proprietary legend. The intention is to keep the program secret. The copyright marking is meant to provide protection only in the event this fails, and the pro-

---

186. Id. at 895-96.
187. Id.
191. Id.
gram markings may so state. Once the owner elects to utilize copyright protection, he or she may presumably register and deposit copies.

It can be argued that disseminating with copyright notice, even in a confidential relationship and where accompanied by a proprietary notice, constitutes "publication" and thereby strips the program of its trade secret status. An ancillary and more fundamental question goes to whether seeking this type of bifurcated protection may be inherently inconsistent with public policy. One major purpose of the copyright law is to promote disclosure. The question may be legitimately raised as to whether copyright protection should be countenanced when the owner's real purpose is to thwart disclosure and to rely on his copyright (if one exists) only if this fails and publication occurs as a result of events beyond the owner's control.

In a case involving an allegation that defendants misappropriated plaintiff's computer system reference manual, the document had been marked with both a proprietary legend and a statutory copyright notice. The district court held that the act of affixing the copyright notice did not, as a matter of law, defeat the trade secret claim. The court of appeals, dealing only with the argument of estoppel (and not treating the Constitutional and policy arguments which might be made), agreed.

4. Advantages and Disadvantages of Trade Secrets

Among the advantages of trade secret protection of programs are: (1) the real possibility of securing preliminary relief in the event of litigation; (2) its clear applicability; (3) its applicability to intermediate (flow chart) and ancillary (documentation) materials; (4) the broad scope of protection (concept, disclosure, and use); (5) the relatively limited opportunity for misuse, because generally there is not uncontrolled proliferation; (6) the possibility of lengthy duration; and (7) the rapidity with which protection can attach (immediately).

There are, however, several disadvantages to trade secret protection of programs. One disadvantage, from the viewpoint of society if not the developer, is the unfortunate effect which the resultant "cesspool of secrecy" may have on progress in the field. A second disadvantage is that secrecy is not well suited to a product for which wide

193. Id. at 347.
proliferation is sought. Indeed, there may be inherent in the definition of a trade secret the notion that it not be known to most competitors of the owner or user.\textsuperscript{195} However, in one case dealing squarely with that issue (where software was the \textit{res} in question),\textsuperscript{196} the court saw no infirmity arising by virtue of mere proliferation. A third disadvantage is the possibility of an immediate loss of protection, should the secret become public. A fourth disadvantage is that affirmative steps must be taken to maintain the trade secret. Finally, the possibility of pre-emption is a recognizable disadvantage.

VI. THE SHRINK WRAP LICENSE

The "shrink wrap" is a phenomenon which has arisen in response to an attempt to protect relatively inexpensive, mass marketed software. A brief historical summary will be helpful in describing how certain pertinent aspects of the software industry have changed, and given rise to such a need.

In the early days of computing, a typical mainframe computer cost hundreds of thousands (and perhaps over a million) dollars. The application software was all custom developed, and a typical program might cost tens of thousands (or even more than a hundred thousand) dollars. Then, slowly, the concept of the program package made inroads. However, a typical program package for a mainframe computer might cost on the order of $50,000. Legal protection in those days was provided almost exclusively by the law of trade secrets, with a written license contract signed by the parties. It was drafted by lawyers and often the transaction was negotiated by lawyers.

The key elements were that each such transaction was substantial and that the parties were corporations, to whom contracts and lawyers were a way of life.

However, the technology advanced rapidly. The price of hardware plummeted and the industry saw the introduction of the personal computer. Today, most computers in existence cost less than $10,000; indeed many cost less than $2,000. As the price of hardware fell, increasing numbers of computers were sold. As to certain models, there are presently hundreds of thousands of units. On some models, there are more than one million units extant. Today a copy of a typical

\begin{footnotesize}
\begin{enumerate}
\item[195.] The Restatement of Torts lists six factors for use in the determination as to whether a trade secret exists. One is the extent to which the subject matter is known outside the enterprise; another is the ease with which it may properly be acquired by others. \textit{Restatement of Torts} § 757 comment 10, at 6 (1939).
\end{enumerate}
\end{footnotesize}
personal computer program may be marketed at a price in the range of fifty to perhaps eight hundred dollars. These are marketed over the counter like expensive jelly beans. At these prices, there are no lawyers; moreover, even asking the buyer to sign a form contract is viewed by many as imposing an impediment to sales.

In this environment, how does the software developer protect his investment? One proposed solution is the shrink wrap (also called the boxtop, tear-open, or blister) license. The programming device on which the copy of the program is embodied is a floppy disk\(^{197}\) or a ROM\(^{198}\) which comes encased in a clear plastic wrapper. Visible through this wrapper is a sheet of paper bearing, in large print, a notice stating that whoever breaks the wrapper\(^{199}\) thereby agrees to a list of stated conditions.

A question sometimes asked is why the marketer bothers with such a procedure. Why not simply copyright the program and let it go at that? Aside from the fact that the scope of copyright protection is not well established, the answer generally given is that there are elements of protection which can probably not be secured by copyright alone and which many software developers view as valuable. For example, the copyright law would not prohibit the owner of a copy, in the absence of an agreement to the contrary, from selling or leasing that copy. Or, the developer may wish to limit the use of each copy to a particular computer or terminal. Copyright alone will not do that. Furthermore, the developer may wish to protect the trade secrets embodied in the program against reverse engineering and disassembly.\(^{200}\) Again, copyright alone may not serve the purpose.

Accordingly, many vendors now sell copies of their programs accompanied by a shrink-wrap license. Possible problems arise in connection with this practice. For example, a valid contract requires offer and acceptance, and there is a question as to whether this has been achieved in the typical shrink-wrap situation. The buyer presumably paid his consideration at the time he purchased the copy, which is before he opened the wrapper or otherwise purportedly activated the terms of the shrink wrap license. However, to the extent there has

\(^{197}\) A floppy disk looks like a 45 rpm phonograph record, and records the program on one or both surfaces in the form of magnetic impulses. It is similar to a hard disk, except that the floppy disk is generally smaller and flexible.

\(^{198}\) The actual tiny silicon chip comprising the ROM is embodied in a plastic casing about the size of a cigarette pack.

\(^{199}\) Alternatively, the notice may key acceptance to breaking an inner wrapper (which contains only the programming device), or using the programming device.

\(^{200}\) See Gordon & Starr supra, note 6 at 501.
been any acceptance of those terms, presumably it took place only when he opened the shrink wrap. Is it possible to convert a sale into a license?

Moreover, there appears to be a modern trend in consumer oriented law which makes it difficult to enforce "adhesion contracts" where the result is deemed unfair. Would that pose a problem for the shrink wrap license? Further, a litigation problem is posed by the possible difficulty of proving notice. Since many purchasers of copies of personal computer programs have not yet reached the age of majority, they may not possess the capacity to enter into a contract. There appear to be no cases on point as yet.

VII. INTERNATIONAL SOFTWARE PROTECTION

There are two ways of seeking software protection abroad. One is by using an international copyright convention or treaty and the other is by using the domestic law of the country in question.

A. By Convention

There are two important multilateral copyright conventions. The Berne Convention (formally the International Union for the Protection of Literary and Artistic Works) is administered by the World Intellectual Property Organization (WIPO).201 Although the United States has never joined the Berne Convention, American nationals can secure its protection through publication in a Berne country simultaneously with publication in the United States.202 The Universal Copyright Convention is administered by UNESCO.203 Under this treaty,

201. As of January 1, 1984, the following nations had ratified the Berne Convention: Argentina, Australia, Austria, Bahamas, Barbados, Belgium, Benin, Brazil, Bulgaria, Cameroon, Canada, Central African Republic, Chad, Chile, Congo, Costa Rica, Cyprus, Czechoslovakia, Denmark, Egypt, Fiji, Finland, France, Gabon, German Democratic Republic, Germany, Federal Republic of, Greece, Guinea, Holy See, Hungary, Iceland, India, Ireland, Israel, Italy, Ivory Coast, Japan, Lebanon, Libya, Liechtenstein, Luxembourg, Madagascar, Mali, Malta, Mauritania, Mexico, Monaco, Morocco, Netherlands, New Zealand, Niger, Norway, Pakistan, Philippines, Poland, Portugal, Romania, Rwanda, Senegal, South Africa, Spain, Sri Lanka, Surinam, Sweden, Switzerland, Thailand, Togo, Tunisia, Turkey, United Kingdom, Upper Volta, Uruguay, Venezuela, Yugoslavia, Zaire and Zimbabwe. 4 M. NIMMER, THE LAW OF COPYRIGHT App. 22 (1984).


203. As of January 1, 1984 the following nations had ratified the Geneva Act of 1952 and the Paris Act of 1971. The following chart has been reprinted with permission from 4 M. NIMMER, THE LAW OF COPYRIGHT App. 22 (1984), Copyright © 1984 by Matthew Bender & Co., Inc.

|------------------|---------------------|------------------|
works of U.S. nationals wherever published must be granted by all

<table>
<thead>
<tr>
<th>Country</th>
<th>Date</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Algeria</td>
<td>August 28, 1973</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Argentina</td>
<td>February 13, 1958</td>
<td>August 14, 1982</td>
</tr>
<tr>
<td>Australia</td>
<td>May 1, 1969</td>
<td>December 27, 1976</td>
</tr>
<tr>
<td>Austria</td>
<td>July 2, 1957</td>
<td>August 5, 1975</td>
</tr>
<tr>
<td>Bahamas</td>
<td>August 5, 1975</td>
<td>June 18, 1983</td>
</tr>
<tr>
<td>Bangladesh</td>
<td>June 18, 1983</td>
<td>June 18, 1983</td>
</tr>
<tr>
<td>Barbados</td>
<td>August 31, 1960</td>
<td>December 1, 1982</td>
</tr>
<tr>
<td>Belgium</td>
<td>January 13, 1960</td>
<td>December 11, 1975</td>
</tr>
<tr>
<td>Brazil</td>
<td>June 7, 1975</td>
<td>June 7, 1975</td>
</tr>
<tr>
<td>Bulgaria</td>
<td>May 1, 1973</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Cameroon</td>
<td>August 10, 1962</td>
<td>April 17, 1980</td>
</tr>
<tr>
<td>Cameroon</td>
<td>August 10, 1962</td>
<td>April 17, 1980</td>
</tr>
<tr>
<td>Chile</td>
<td>September 16, 1955</td>
<td>January 14, 1956</td>
</tr>
<tr>
<td>Colombia</td>
<td>June 18, 1976</td>
<td>March 7, 1980</td>
</tr>
<tr>
<td>Costa Rica</td>
<td>September 16, 1955</td>
<td>December 10, 1980</td>
</tr>
<tr>
<td>Cuba</td>
<td>June 18, 1957</td>
<td>December 10, 1980</td>
</tr>
<tr>
<td>Czechoslovakia</td>
<td>January 6, 1960</td>
<td>April 17, 1980</td>
</tr>
<tr>
<td>Democratic Kampuchea</td>
<td>September 16, 1955</td>
<td>September 16, 1955</td>
</tr>
<tr>
<td>Denmark</td>
<td>February 9, 1962</td>
<td>July 11, 1979</td>
</tr>
<tr>
<td>Dominican Republic</td>
<td>May 8, 1983</td>
<td>May 8, 1983</td>
</tr>
<tr>
<td>Equador</td>
<td>June 5, 1957</td>
<td>March 29, 1979</td>
</tr>
<tr>
<td>El Salvador</td>
<td>March 29, 1979</td>
<td>March 29, 1979</td>
</tr>
<tr>
<td>Fiji</td>
<td>October 10, 1970</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Finland</td>
<td>April 16, 1963</td>
<td>December 10, 1980</td>
</tr>
<tr>
<td>France</td>
<td>January 14, 1956</td>
<td>December 10, 1980</td>
</tr>
<tr>
<td>German Democratic</td>
<td>October 5, 1973</td>
<td>September 16, 1955</td>
</tr>
<tr>
<td>Republic</td>
<td>October 5, 1973</td>
<td>September 16, 1955</td>
</tr>
<tr>
<td>Germany, Federal Republic</td>
<td>September 16, 1955</td>
<td>September 16, 1955</td>
</tr>
<tr>
<td>Greece</td>
<td>August 24, 1963</td>
<td>November 13, 1981</td>
</tr>
<tr>
<td>Guinea</td>
<td>November 13, 1981</td>
<td>November 13, 1981</td>
</tr>
<tr>
<td>Haiti</td>
<td>September 16, 1955</td>
<td>November 13, 1981</td>
</tr>
<tr>
<td>Holy See</td>
<td>October 5, 1955</td>
<td>May 6, 1980</td>
</tr>
<tr>
<td>Hungary</td>
<td>January 23, 1971</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Iceland</td>
<td>December 18, 1956</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>India</td>
<td>January 21, 1958</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Ireland</td>
<td>January 20, 1959</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Israel</td>
<td>September 16, 1955</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Italy</td>
<td>January 24, 1957</td>
<td>January 25, 1980</td>
</tr>
<tr>
<td>Japan</td>
<td>April 28, 1956</td>
<td>October 21, 1977</td>
</tr>
<tr>
<td>Kenya</td>
<td>September 7, 1966</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Laos</td>
<td>September 16, 1955</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Lebanon</td>
<td>October 17, 1959</td>
<td>October 17, 1959</td>
</tr>
<tr>
<td>Liberia</td>
<td>July 27, 1956</td>
<td>October 17, 1959</td>
</tr>
<tr>
<td>Liechtenstein</td>
<td>January 22, 1959</td>
<td>October 17, 1959</td>
</tr>
<tr>
<td>Luxembourg</td>
<td>October 15, 1955</td>
<td>October 15, 1955</td>
</tr>
<tr>
<td>Malawi</td>
<td>October 26, 1965</td>
<td>October 26, 1965</td>
</tr>
<tr>
<td>Malta</td>
<td>November 19, 1968</td>
<td>November 19, 1968</td>
</tr>
</tbody>
</table>
signatories at least as much copyright protection as that state accords works of its own nationals first published there.204

A French court has ruled that, aside from whether French domestic copyright law applies, a videogame registered with the U.S. Copyright Office is protected in France by virtue of Article II of the Universal Copyright Convention, signed by the United States and France: "... published works of nationals of any contracting state and works first published in that state shall enjoy in each contracting state the same protection as that other state accords in its own territory, as well as the protection specifically granted by the Convention."205

One question pertinent to the program milieu is whether this obligation applies to rights not contained in the state's copyright laws, but which may be rights in the nature of copyright. More specifically, if a foreign country's copyright law does not permit protection of programs, but its trade secret, patent, or other laws provide rights in the nature of copyright protection for the programs, can a U.S. national

<table>
<thead>
<tr>
<th>Country</th>
<th>Date of Ratification 1</th>
<th>Date of Ratification 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mauritius</td>
<td>March 12, 1968</td>
<td></td>
</tr>
<tr>
<td>Mexico</td>
<td>May 12, 1957</td>
<td>October 31, 1975</td>
</tr>
<tr>
<td>Monaco</td>
<td>September 16, 1955</td>
<td>December 13, 1974</td>
</tr>
<tr>
<td>Morocco</td>
<td>May 8, 1972</td>
<td>January 28, 1976</td>
</tr>
<tr>
<td>Netherlands</td>
<td>June 22, 1967</td>
<td></td>
</tr>
<tr>
<td>New Zealand</td>
<td>September 11, 1964</td>
<td></td>
</tr>
<tr>
<td>Nicaragua</td>
<td>August 16, 1961</td>
<td></td>
</tr>
<tr>
<td>Nigeria</td>
<td>February 14, 1962</td>
<td></td>
</tr>
<tr>
<td>Norway</td>
<td>January 23, 1963</td>
<td>August 7, 1974</td>
</tr>
<tr>
<td>Pakistan</td>
<td>September 16, 1955</td>
<td>September 3, 1980</td>
</tr>
<tr>
<td>Panama</td>
<td>October 17, 1962</td>
<td></td>
</tr>
<tr>
<td>Paraguay</td>
<td>March 11, 1962</td>
<td></td>
</tr>
<tr>
<td>Peru</td>
<td>October 16, 1963</td>
<td></td>
</tr>
<tr>
<td>Phillippines</td>
<td>November 19, 1955</td>
<td></td>
</tr>
<tr>
<td>Poland</td>
<td>March 9, 1977</td>
<td>March 9, 1977</td>
</tr>
<tr>
<td>Portugal</td>
<td>December 25, 1956</td>
<td>July 30, 1981</td>
</tr>
<tr>
<td>Senegal</td>
<td>July 9, 1974</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Soviet Union</td>
<td>May 27, 1973</td>
<td></td>
</tr>
<tr>
<td>Spain</td>
<td>September 16, 1955</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Sweden</td>
<td>July 1, 1961</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Switzerland</td>
<td>March 30, 1956</td>
<td></td>
</tr>
<tr>
<td>Tunisia</td>
<td>June 19, 1969</td>
<td>June 10, 1975</td>
</tr>
<tr>
<td>United Kingdom</td>
<td>September 27, 1957</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>United States of America</td>
<td>September 16, 1955</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Venezuela</td>
<td>September 30, 1966</td>
<td></td>
</tr>
<tr>
<td>Yugoslavia</td>
<td>May 11, 1966</td>
<td>July 10, 1974</td>
</tr>
<tr>
<td>Zambia</td>
<td>June 1, 1965</td>
<td></td>
</tr>
</tbody>
</table>

204. 4 M. Nimmer, supra note 201, § 1704[B].

claim such protection by virtue of the convention?\textsuperscript{206}

B. \textit{Foreign Domestic Law}

Based on reports believed reliable, the following sets forth, if in somewhat fragmentary form, the existing and prospective domestic law of program protection for a number of foreign countries.\textsuperscript{207} In general, the application of law to the program scenario is less well developed abroad that in the United States.

1. Japan\textsuperscript{208}

\textit{Taito v. ING Enterprises}\textsuperscript{209} was an action for infringement of a copyright on a program written in assembly language. The alleged infringement constituted copying a ROM embodying the program.\textsuperscript{210}

\textsuperscript{206} To the extent such other laws do not distinguish between nationals and foreigners, perhaps nothing turns on the answer to this question.

\textsuperscript{207} The following section on foreign domestic law is based, in part, on various materials, believed to embody accurate information, which have come to the author's attention. It is designed to serve merely as a survey of current software protection law developments in progress. Where possible, the author has provided complete citations to foreign case law, statutory law, and secondary sources.

\textsuperscript{208} The information on Japan has been derived from the English Edition of Yamasaki, \textit{Japanese Case Law Report}, AIPPI J. 107 (Sept. 1983).

\textsuperscript{209} \textit{Id.} at 107-08. Tokyo Dist. Ct., Dec. 6, 1982.

\textsuperscript{210} \textit{Id.} at 108-09. The Japanese copyright act provides in part:

\begin{quote}
\textquotedblleft (Definitions)
\textbf{Article 2}
\textquotedblleft (1) The terms in each of the following items shall have the meaning as defined in each item as far as the present Law is concerned:
\begin{itemize}
\item 1. Work — Creative expression of thoughts or sentiments which falls within the literary, scientific, artistic or musical domain.
\item 15. Reproduction — Remaking in a tangible form by means of printing, photography, copying, sound or visual recording and other methods \textup{[}the remainder omitted\textup{]}
\end{itemize}
\textbf{Article 10}
\textquotedblleft (1) The works of the present Law shall comprise the following by way of example:
\begin{itemize}
\item 1. novels, dramas, articles, lectures and other literary works
\item 2. works of music
\item 3. works of choreography or pantomime
\item 4. paintings, engravings, sculptures and other works of art
\item 5. works of architecture
\item 6. works of cartography and other figurative works of scientific nature, such as plans, charts and models
\item 7. works of cinematography
\item 8. works of photography
\end{itemize}
\end{quote}
The issues were whether a program may constitute a copyrightable work and whether an object program embodied in ROM constitutes a copy.\footnote{Id.}

Defendant argued that this program was not copyrightable because the symbolic source language was not readily comprehensible to people and, therefore, was not a means for expressing thought. The court held that the program was indeed copyrightable and that the ROM embodying the object code constituted a copy. The object of the program was to cause a display of a video game, and the expression in symbolic language was comprehensible to persons with expert knowledge. Different programs could produce the same result and the program was seen by the court as a creative expression of thought, and copyrightable. The court noted that the object program embodied in ROM was a mechanical conversion of the source program. No creativity was involved in converting one into the other. Creating such a ROM constituted making a copy of the source program.

On the legislative front, a subcommittee of the Copyright Council

\footnote{Id.}
(an advisory body to the Education Ministry) has proposed a fifty year term of copyright for programs. The Ministry for Posts and Telecommunications has also expressed an interest in the matter. However, the Ministry for International Trade and Industry (MITI) has proposed a fifteen year term (the same as for Japanese patents), with authority in the government to require compulsory licensing for a fee. The MITI proposal was withdrawn in 1984 but is expected to resurface in spring, 1985. In the United States, the Computer business Equipment Manufacturers Association (CBEMA) stated that its members would reevaluate their plans in Japan if the MITI proposal were adopted; CBEMA has urged that Congress consider reciprocal measures.

2. Germany

a. Patents

Prior to 1974, German courts permitted patents on certain programmable processes and programmed machines. In 1974, the German courts did an about face, and they have been negative ever since. Indeed, German courts have enunciated a number of negative rules, but no positive rules for patentability.

In 1978, the Federal Republic of Germany enacted a new patent act in accord with the European Patent Convention. In the Act, programs are expressly excluded as such from patentability.

b. Copyright

VisiCorp brought an action in the Munich trial court against a group of German companies, alleging that they copied its VisiCorp program.212

Section 2 of the German Copyright Act lists, *inter alia*, literary linguistic works, or representations of a scientific or technological nature. Under German copyright law, a work within one of these categories, which is a personal intellectual creation, constitutes copyrightable subject matter. The court held that if the task permitted several different solutions, then the creative intellectual content of a program would be expressed by the choice, collection, review, arrangement, and classification of the material. The court further held that programs meet the linguistic requirement despite the fact that programs are written in a special computer language and can be read only by special means.

Moreover, the court ruled that a program also meets the scientific and technical requirement and that the intellectual-aesthetic content could be found not only in the visually perceptible form, but also "in the content of ideas which can be expressed in the formation and sequence of thoughts of the represented concepts and/or in the special intellectual form and kind of collection, classification and arrangement of the offered material."213

In VisiCorp, with a relatively complex program with about 10,000 instructions, the court has ample opportunity for personal creative formation entering into the program itself. Accordingly, the court held it copyrightable under Section 2.214 The decision is on appeal, with a ruling expected in 1985. The trial court in Cassel and the trial court in Mosbach have also held programs copyrightable. However, these decisions conflicted with a 1981 decision handed down by a trial court in Mannheim.

Several laws pertaining to software protection are being considered. One would proscribe illegal copying of software, and another would make theft of programs a crime. According to an official in the German Ministry of Justice,

there is no question that the loading of the program into the computer system is a copying of the program. But what about the running of the program? . . . In my opinion, the answer has to be in the affirmative. When running the program, the transfer of each bit of information from one storage location to another within the computer is copying the program.215

3. Canada216

As of summer, 1984, there were several pending cases involving copyrightability of programs in Canada, but no decisions.

In May, 1984 the Canadian government released a "White Paper" proposing revisions to the Copyright Act. Among the proposals was one which would protect machine-readable data bases and computer programs and data bases for five years. Programs in human-readable form would, as in the past, receive protection similar to that

213. Id.
214. Id.
afforded traditional literary works for fifty years plus life, with the owner having the right to prevent making machine-readable programs. If it is the human-readable program that is published, this right would terminate five years after the end of the year of creation of the human-readable program in human-readable form.

For an unpublished program in machine-readable form, the copyright holder would have the right to publish in machine-readable form, to reproduce, and to make a human-readable program based on it. However, he or she would not have the right to rent or use. Under this proposal, selling, leasing, licensing, trading, or offering to do any of these would be an act of publication. An unpublished machine-readable program would be protected for five years from creation. If published, protection would run until five years after the end of the year of publication. However, if published more than five years after creation, the program would be ineligible for copyright protection.

4. United Kingdom

a. Patent

In the U.K., trade secret, contract, copyright (without registration or publication requirement), and patent are available to protect programs.

In 1979, the U.K. passed a new Patent Act to bring about accord with the European Patent Convention and no pertinent decisions are yet available under the new act, although patents have regularly been granted for program implemented processes and systems. The new contributory infringement section of the Patent Act may well apply to the distribution of computer programs which would produce infringement when used.

There are at least two organizations in the U.K. devoted to restraining the activities of computer software pirates: Federation Against Software Theft (FAST), and Federation Against Copyright Theft (FACT).

b. Copyright

Although the Copyright Act of 1956 makes no specific reference to programs, a number of cases have stated that programs would be protected as "literary works." In several cases granting ex parte applications to seize infringing goods, courts ruled that object code was a

---

218. Copyright Act, 1956, 4 & 5 Eliz. 2, Ch. 74
reproduction or an adaptation of source code.\textsuperscript{219}

In a form not visible to humans, programs would probably still qualify as literary works, even though they would not be "writings." Published and unpublished works are protected by the act, and registration is not required. Copyright inheres at the time the unpublished literary work is made. It inheres in published works on first publication.

c. Interaction of Copyright and Trade Secret

Sometimes the owner may seek both copyright and trade secret protection for a program. Software used only in-house may be copyrighted under section 2(1) of the 1956 Act as an unpublished work and may also be protected by confidentiality clauses. Here the principles of copyright and trade secret do not conflict.

However, where copyright attaches under section 2(2) by virtue of publication, it is uncertain whether trade secret protection may also be claimed. Whether trade secret can be claimed will depend on the type of publication. English law is unclear as to what degree of publication is necessary before the information ceases to be protectable by trade secret. However, confidentiality is not stripped away merely where programs are marketed widely, even though capable of reverse engineering. The question then becomes how much work has to be done to accomplish the reverse engineering. In the case of multiple disclosures such as to licensees, the law is less likely to impose an obligation of confidence where none is specifically set forth in the license agreement.

Publication is insufficient to cause copyright to attach under section 2(2) unless it is "intended to satisfy the reasonable requirements of the public."\textsuperscript{220} Placing a program on general sale to computer owners does not necessarily end any claims to confidentiality. The program may be protected as a trade secret if significant effort would be required to reverse engineer or copy the program. Publication to licensees would not necessarily end confidentiality of the software, especially if the license agreement contained a confidentiality clause. Many licensed programs would be protected under both the Copyright Act and trade secret law.


\textsuperscript{220} Copyright Act 1956, § 49 (2)(b).
5. France\textsuperscript{221}

French law offers six methods which may be used to secure software protection.

Although French patent law excludes patentability for a computer program as such,\textsuperscript{222} a system or an embodied process may be patented. French design protection can be used to protect video games and other visual displays resulting from a program. It does not protect utilitarian objects when their designs are totally dependent on their utilitarian character. Registration of a picture or drawing is required, and the protection period is twenty-five years, renewable for another twenty-five years. At the depositor's request, the deposit will be kept secret.

While French copyright law does not include programs expressly, a number of decisions have protected programs embodied in ROM or on a tape or diskette, whether operating systems or application programs, and whether in source or object code. An original literary or artistic work accrues copyright protection at creation, and registration is not required.

A program is original if the result of an intellectual creative process and if the program itself does not follow directly from the goal set by the programmer (i.e., there must be choices among various programming steps).

French law recognizes protection through total or partial secrecy. In the case of total secrecy (where the program is disclosed to no one outside the enterprise), one who steals it through an employee is liable under the criminal law. In the case of partial secrecy (where there is a license with non-disclosure agreement), the civil law provides a remedy. Secrecy does not preclude protection by copyright law.

Among the remedies provided by criminal law are unlawful disclosure of a trade secret,\textsuperscript{223} disclosure of a professional secret,\textsuperscript{224} abuse of confidence,\textsuperscript{225} corruption,\textsuperscript{226} and theft.\textsuperscript{227} A decision under Article 418 held that it applied to programs; applicability had been unclear.

\textsuperscript{222} See Article 6 of the French Patent Law (adapted to Article 52 of the European Patent Convention).
\textsuperscript{223} Penal Code Art. 418.
\textsuperscript{224} Art. 378.
\textsuperscript{225} Art. 408.
\textsuperscript{226} Art. 177-79.
\textsuperscript{227} Art. 379.
because courts traditionally did not apply this provision to services.228

The civil law recognizes causes of action for misappropriation of intellectual work and for unjust enrichment.

When several employees collaborate to produce a work, and the contribution of each cannot be separated from the whole, the law deems it a "collective work," which belongs to the employer.

There have been a number of projects directed toward the issue of how software should be protected. In August, 1983, the Minister of Industry established a commission under the guidance of the patent office to report on the present status and to propose changes. A report presented in December, 1983, suggested continuing present law but adding a new system of protection specifically directed to computer programs and providing twenty years of protection for algorithms of registered programs.

6. Australia

In Apple Computer, Inc. v. Computer Edge Pty, Ltd.,229 the trial court denied protection, finding that computer programs were not within the subject matter copyrightable under Australian law. The appellate court reversed. In light of the confusion, the Copyright Act of 1968 was amended. However, the new act does not specifically extend protection to machine-readable software.230

228. SESA c.v. TC NANTERRE, 5 Mai 1981, reported in Expertises 1983, No. 57 at 579.
230. Copyright Amendment 1984, No. 43:
1. adding a definition of "computer program" to of the Act as follows:
   "'computer program' means and expression, in any language, code or notation, of a set of instructions (whether with or without related information) intended, either directly or after either or both of the following:
   (a) conversion to another language, code or notation;
   (b) reproduction in a different material form, to cause a device having digital information processing capabilities to perform a particular function";
2. adding to the existing category or "literary works" in Section 10 the following:
   "(a) a table, or compilation expressed in words, figures or symbols (whether or not in a visible form);
   (b) a computer program or compilation of computer program";
3. Specifically defining an "adaptation" of a computer program to include a translation from one language to another, i.e., "a version of the work (whether or not in the language, code or notation in which the work was originally expressed) not being a reproduction of the work" (Section 10);
4. redefining the term "infringing copy" in section 10 to include copies of adaptations of works and to remove an anomaly whereby software imported with permission might nevertheless technically be considered an infringing copy;
5. defining "material form" in Section 10 to include "any form (whether visible or
7. South Africa

In a 1981 case, plaintiff's employee developed an accounting package for professionals, but refused to turn over the source code to his employer.231 The court held that under the applicable 1978 Copyright Act, these programs were copyrightable subject matter, and that the employee owned them.232 The court did not proscribe his redevelopment of similar programs ab initio, even were he to use the knowledge he had gained. Certain language in the case suggests that data bases are not protectable under South African copyright law.

8. Brazil

Brazil appears ready to create a system of legal protection unique

not) of storage from which the work or adaptation, or a substantial part of the work or adaptation, can be reproduced";

6. creating a presumption that the making of a back-up copy of a computer program is not an infringement by adding a new Section 43A, which states:

"43A. (1) Subject to sub-section (2), the copyright in a literary work being a computer program is not infringed by the making of a reproduction of the work, or of a computer program being an adaptation of the work, if—
(a) the reproduction is made by, or on behalf of, the owner of the copy (in this Section referred to as the 'original copy') from which the reproduction is made; and
(b) the reproduction is made for the purpose only of being used, by or on behalf of the owner of the original copy, in lieu of the original copy in the event the original copy is lost, destroyed or rendered unusable.
(2) Sub-section (1) does not apply to the making of a reproduction of a computer program, or of an adaptation of a computer program—
(a) from an infringing copy of the computer program; or
(b) contrary to an express direction by or on behalf of the owner of the copyright in the computer program given to the owner of the original copy not later than the time when the owner of the original copy acquired the original copy.
(3) For the purposes of this section—
(a) a reference to a copy of a computer program or of an adaptation of a computer program is a reference to any article in which the computer program or adaptation is reproduced in a material form; and
(b) a reference to an express direction, in relation to a copy of a computer program or of adaptation of a computer program, includes a reference to a clearly legible direction printed on the copy or on a package in which the copy is supplied";

7. extending "the anti-piracy" provisions of the current law (Section 132) to cover creation of infringing copies by telephone or radio transmission of a program;
8. Creating an offense for advertising the supply of infringing copies of a program;
9. conferring copyright protection on existing programs; and,
10. exempting actions that took place prior to the commencement of the amendments from being considered infringements.

232. A 1980 amendment apparently has altered the situation so that the employer would own programs written after May 22, 1980.
to computer programs. Under the draft being circulated in late summer, 1984, the law would require registration of source and object code as a condition for protection, and would grant exclusive rights for ten years. All documentation, including source code, would be maintained in sealed form by the government to insure free use upon expiration.

To maintain the registration, the producer would be compelled to provide technical assistance to users during the registration period, and in the event of national urgency, would be required to grant third parties the right to reproduce. Compulsory licensing of Brazilian companies would be required. In addition, there would be a requirement for active use of the software, possibly within two years of registration.

9. Hungary

"Authorities" have determined that software is covered under the country's existing copyright law as a writing, albeit a special type. Hungary favors utilizing copyright protection, rather than creating sui generis protection for software, both because it would require time to develop a unique system and because such protection would be limited to a single country without possibility of similar international protection.

10. Taiwan

The Taiwanese legislature is expected to receive in 1984 a revised copyright bill which would explicitly extend protection to programs.

11. Mexico

Mexico began accepting programs for copyright registration in October, 1984. Mexico appears to be closely watching events in Brazil.

233. Such a conclusion seemed apparent by the end of an international seminar on legal protection of software, held in Brasilia in early summer 1984. The seminar was under the auspices of the Special Secretariat for Information, which was expected to propose such a software protection law by the end of 1984.

234. This requirement is similar to that found in Brazilian patent law.


236. See supra note 207 and accompanying text.

237. 28 Pat. Trademark & Copyright J. (BNA) 482 (Aug. 30, 1984), Report of Gyorgy Palos, Director of the Legal Department, Bureau of Copyright Protection.

238. See supra note 207 and accompanying text.

239. Id.
12. Spain

Spain is amending its copyright law.\textsuperscript{240} Despite internal pressure to dilute software protection, Spain must strengthen intellectual property protection as a prerequisite for entry into the European community.

13. Colombia

Based on reports believed reliable,\textsuperscript{241} Colombia is closely watching events in Brazil with respect to developing domestic laws for program protection.

14. Malaysia

The government intends\textsuperscript{242} to amend its copyright law, strengthening protection. Domestic manufacturers seek increased protection so as to better participate in joint ventures with foreign software manufacturers. However, the National Computer Board is looking seriously at the Japanese MITI approach.

15. Singapore

 Reliable reports\textsuperscript{243} indicate that Singapore, like Malaysia, intends to amend its copyright law to strengthen program protection.

16. Thailand

Thailand, like Malaysia and Singapore, is preparing to amend its copyright law to strengthen program protection.\textsuperscript{244}

17. Sweden

Sweden is considering legislation similar to U.S. copyright legislation.\textsuperscript{245}

18. Netherlands

The Netherlands is considering legislation similar to U.S. copyright legislation.\textsuperscript{246}

\textsuperscript{240} Id.
\textsuperscript{241} Id.
\textsuperscript{242} Id.
\textsuperscript{243} Id.
\textsuperscript{244} Id.
\textsuperscript{245} Id.
\textsuperscript{246} Id.
VIII. EXTRALEGAL MEANS FOR PROTECTING PROGRAMS

Legal means are but one vehicle for seeking to protect programs, and there are certain other means which may be at least as effective in a given situation. In addition to the suggestions in this section, some of those set forth below247 bear on extralegal methods of protection. Basically, extralegal means of achieving protection are based on structuring the commercial environment in such a way as to reduce the incentive to misappropriate, and the use of technology so as to render it more difficult to misappropriate.

Commercially, there are two steps which some licensors may take to protect their investment in programs. The first is to provide maintenance for their licensed programs. Although the licensor generally undertakes significant (sometimes massive) debugging activities before licensing, the typical licensed program is an extremely complex item, containing numerous bugs which will manifest themselves on a continuing basis for years. Any one of these is capable, under the right (more appropriately, the wrong) set of circumstances, of immobilizing a licensee’s efforts when a bug causes the program to cease running or to run improperly. Most licensees will, therefore, be willing to pay an additional fee for licensor maintenance (sometimes called “support”). In providing this support, the licensor agrees, on a timely basis, to remedy any bug uncovered by the licensee. Some licensors market the support under the basic licensing agreement, while others may market it as an additional optional item. Yet others offer no support.

A second commercial offering which may afford some protection is providing program enhancements. Many programs are such that it is economically feasible to improve them as time progresses. For example, new options may be added to increase the number of functions which the program can perform. Furthermore, changes may be made in certain algorithms to reduce running time. Where enhancements are to be supplied, provision is generally made in the licensing agreement.

The value of a program without support or right to enhancement (this is what a misappropriator generally gets) is often far less than that of a supported program with enhancement rights. Accordingly, the incentive to misappropriate the bare program is reduced somewhat.

Technologically, one might alter either the software or the hardware in such a manner that each physical programming device, or

247. See infra note 252 and accompanying text.
copies of it, will operate only with a specific computer. For example, provision might be made in the program to incorporate a comparison between the serial number of the computer on which the program is running\textsuperscript{248} and the serial number of the computer for which the program is licensed\textsuperscript{249}. This technique can cause complications and is often not done today. One complication is that such a program will not run on a "back-up" computer in the event that the licensee's computer fails to function.

Another technological means for thwarting piracy is a microprocessor chip which can be inserted in the computer to provide a deciphering function. The program itself is encrypted before being provided to the licensee, and the chip is designed to decipher the encrypted program. The encrypted program is useless without the chip.\textsuperscript{250}

None of these technological methods is foolproof, and each simply increases the difficulty, cost, or risk which a would-be pirate must incur.

IX. CONTRACTING SUGGESTIONS

A. Proprietary Suggestons

The license generally is restricted to a single particular computer (identified in the agreement by serial number). If use on more than one computer is desired, the additional computers are treated as calling for additional licenses (often at reduced royalties). One exception commonly found in licenses is that in the event the identified computer fails to function properly, use may be made on a "back-up" computer. Limiting use to a single computer is regarded as an important method of limiting improper proliferation.

The license will specify the form of the program to be provided to the licensee. Generally, the source code form is more valuable to the licensee, and may be priced higher and be subject to more restrictions. If source code is provided, flow charts and listings may also be provided. Alternatively, a licensor may transfer only the object code form, accompanied by as little documentation as possible.

Much of the agreement should deal explicitly with the protection of proprietary rights. The licensee may acknowledge the licensor's ownership of intangible property in the program, be it trade secrets,

\textsuperscript{248} While the program is running on the computer, it can "read" this serial number.

\textsuperscript{249} This serial number can be written into each licensed programming device by the licensor.

\textsuperscript{250} For one such chip, see U.S. Patent No. 4,168,396.
confidential information, patent rights, or copyrights. Where trade secret material is involved, the agreement should explicitly establish a confidential relationship, and may specify that all tangible materials transferred to the licensee remain the licensor's property. The licensee should be obligated not to engage in any use, transfer, or disclosure of licensed materials and information except as expressly set forth in the agreement. He should further be obligated to inform his employees and agents who may come into contact with the materials of their confidential nature, and should agree to take reasonable precautions to prevent any unauthorized use, transfer, or disclosure and, failing this, to report all such instances to the licensor. The licensee should also agree to cooperate with the licensor in the event of any litigation concerning the matter.

The specific uses to which the licensee may put the transferred material should be set forth. In particular, the matter of how much and what type of copying (if any) is to be permitted, should be expressly set forth. The licensee may be obligated to keep records of his copying, and even to number his copies. Of great importance is that each copy bear the appropriate notice, be it a notice that an algorithm is patented, that a program is copyrighted, or of a proprietary interest. All notices on programs should be present in human-readable form on the programming device, and in machine-readable form at the beginning and end of the program (and perhaps at certain intermediate locations).

Most licenses do not permit sublicensing, but where it is permitted (as when the licensee is a service bureau operating, for lease on a time-sharing basis, a computer and numerous programs for use on it), the licensee must agree to impose restrictions on his sublicensees if there is any way the latter can obtain access to the program code.

When a program is licensed in source code, the licensee may make modifications (unless prevented by the agreement). Accordingly, it is well for the agreement to state what the parties' rights will be to those modifications. The licensee should also agree that on termination of the license, the licensed program will be deleted from any modification.

Finally, the agreement should set forth the disposition of all licensed materials and copies thereof at termination of the agreement. Return to the licensor may be best, but most licenses provide in the alternative for destruction and certification thereof. Further, any confidentiality provision should survive the agreement.
B. *Other Provisions*

The duration of a license agreement may be perpetual, or may be a fixed term. If enhancements are included, the agreement should probably be for a fixed term so as to accommodate changes in licensing policy.

The licensing fee is often payable as a single lump sum. The programs licensed should be identified (perhaps on an appended schedule), and there may be provision for adding programs in the future. It should be made clear whether the license for the stated fee includes maintenance, enhancements, training, and any modifications necessary to fit the licensee's system. Most program licenses are non-exclusive (otherwise the parties would probably agree to a sale), and most licensors prefer them to be non-transferable so that they have some knowledge as to who has their program, and who is supposed to be paying for its use.

A typical agreement also attempts to resolve the issue of liability in the event of program malfunction (e.g., limited to fee paid; including indirect and consequential damages) and discusses warranty matters (e.g., "as is" without warranty; disclaimer of warranty of merchantability or fitness; warranty only of conformity to printed specifications; warranty acknowledging use). The answer to the question of whether a particular warranty limitation is enforceable will often depend on the particular state in question. The agreement may also include a provision concerning indemnification (including legal expenses) for patent or copyright infringement or trade secret/confidential information misappropriation.

Generally, the licensee may terminate at will, and the licensor may terminate for cause (e.g., breach of agreement, including breach of confidence or failure to pay fees; or indications of bankruptcy). Other matters which may be dealt with in the agreement are responsibility for any applicable taxes, choice of law, and an integration clause.251

X. *Suggested Licenser Procedures*

The licensor should take certain steps to see that its proprietary interests are not compromised. If it seeks patent protection, it must proceed in timely fashion. If it relies on copyright, it must be careful

---

251. Because of the frequency with which misunderstandings seem to permeate computer-related (and specifically software-related) transactions, attempts to introduce parol evidence at trial are quite common.
to place its copyright notice appropriately. In the case of trade secret protection, he should mark all programs and documentation with appropriate legends, and maintain good internal security predicated on distribution on a "need to know" basis, employee confidentiality agreements, notice to employees as to what materials are confidential, and in some cases even secured (guards, fences, etc.) premises. Most importantly, this must all be part of a program, so that the existence of regular procedures can be established in the event of litigation.

In the event of copyright or trade secret protection it may be helpful to introduce into each licensed program a "house mark" consisting of a few trivial and non-functional instructions whose only purpose is to indicate the licensor as the source of the program. The same house mark may be used in all programs, or each package shipped may contain a different one in an attempt to trace any "leak."

In policing against infringement or misappropriation, the licensor will generally find it difficult to identify the instances where a pirate uses the program unlawfully. However, occasional piracy by use only will generally not inflict severe economic harm. Significant injury would result where a pirate attempts to market the program in competition with its rightful proprietor. In that situation, however, the unlawful conduct is much more likely to come to the proprietor's attention. If the proprietor has carefully preserved its rights, it may be able to persuade or compel the pirate to cease and desist, and to remedy any injury.

---