The Test for Proving Copyright Infringement of Computer Software: "Structure, Sequence, and Organization" and "Look and Feel" Cases

George N. Grammas

Follow this and additional works at: http://open.mitchellhamline.edu/wmlr

Recommended Citation
Available at: http://open.mitchellhamline.edu/wmlr/vol14/iss1/4
NOTES

THE TEST FOR PROVING COPYRIGHT INFRINGEMENT OF
COMPUTER SOFTWARE: "STRUCTURE, SEQUENCE,
AND ORGANIZATION" AND "LOOK AND FEEL" CASES

In 1980 Congress enacted statutory changes to the Copyright Act as recom-
mended by the National Commission on New Technology Users of Copyrighted
Works (CONTU). For the first time the Copyright Act explicitly extended
protection to computer software. Computer software infringement has since
become one of the most widely litigated areas of copyright law. Perhaps the
most pressing infringement issue is the separation of idea from expression, par-
ticularly in relation to nonliteral aspects of the computer program—the struc-
ture, sequence, and organization of the program and the look and feel of the
program as perceived by the user.

INTRODUCTION .................................................. 106

I. HISTORY OF COPYRIGHT PROTECTION FOR COMPUTER
SOFTWARE .................................................. 109
   A. Legislative History of the Copyright Act ...................... 109
   B. Case Law: Scope of Copyrightable Subject Matter .......... 114

II. THE TEST FOR PROVING COPYRIGHT INFRINGEMENT OF
COMPUTER SOFTWARE ................................. 117

III. THE STANDARD FOR SEPARATING IDEA FROM EXPRESSION
IN NONLITERAL INFRINGEMENT CASES .................. 123
   A. Expression May Exist in Nonliteral Aspects of Computer
      Software ........................................... 123
   B. Separating Idea from Expression in Structure, Sequence, and
      Organization Cases ............................... 126
   C. Separating Idea from Expression in Look and Feel Cases ... 129
      1. Structure, Sequence, and Organization Factors Applied
         in Look and Feel Cases ......................... 130
      2. Literal or Nonliteral Copying ........................ 131
   D. Combining the Test for Proving Copyright Infringement
      With the Standard for Separating Idea from Expression ... 133

IV. THE STANDARD FOR SEPARATING IDEA FROM EXPRESSION
    IS CONSISTENT WITH POLICY CONSIDERATIONS .......... 134
   A. New Programming Techniques Remain in the Public
      Domain ............................................ 135
   B. Two Instances of Overprotection ............................ 136

CONCLUSION .................................................... 139
INTRODUCTION

There is a class of goods known as "public goods," wherein, the amount of use of the good or service by one person does not reduce the amount available to others if the good has been produced.\(^1\)

An idea is like a "public good" since its availability to the public is never diminished with use. In a utilitarian society, the greatest good would be achieved by the widest possible dissemination of the idea. Each member of society gains satisfaction from the dissemination of the idea, and the usefulness of the idea is not diminished with dissemination.

The policy of the Copyright Act is to promote the widest possible dissemination of ideas while encouraging production of creative and innovative works.\(^2\) The Copyright Act balances these countervailing policies by offering a limited monopoly "in original works of authorship fixed in any tangible medium of expression. . . ."\(^3\) Production of creative and innovative works is encouraged by monopolistic protection of the author's expression of the idea. Dissemination and use of the idea by others, however, is never restricted.\(^4\)

2. Whelan Assoc., Inc. v. Jaslow Dental Laboratory, Inc., 797 F.2d 1222, 1235 (3d Cir. 1986); See also Note, Proposed Judicial Guidelines for Deciding Software Infringement Actions, 32 Wayne L. Rev. 1191, 1196 (1986) (notes these competing policy considerations) [hereinafter Judicial Guidelines].
3. The policy considerations are derived from the constitutional authority for the Copyright Act, promoting useful arts by securing exclusive rights:
   The fundamental policy underlying the Copyright Act is to promote the dissemination of the arts and sciences by providing an incentive for the publication of new works. This is accomplished by granting the original author certain exclusive rights in the work. Under copyright law, the limited monopoly granted must not be so broad that it prevents others from building upon the ideas expressed in the work. Separating ideas from expressions is the primary means of balancing these two competing interests of — providing an incentive to create without allowing an overly restrictive monopoly. Note, Protecting the "Look and Feel" of Computer Software, 1 High Technology L.J. 411, 419 (1986) (footnotes omitted) [hereinafter "Look and Feel"]. "The key is to find a balance that creates an incentive to produce works of authorship, yet promotes the spread of learning, culture, and development." Id. at 420.
4. 17 U.S.C. § 102(a) (1982). The limited monopoly is in the form of exclusive rights to reproduce or distribute the copyrighted work or prepare derivative works. 17 U.S.C. § 106 (1982). "[T]he authorization to grant to individual authors the limited monopoly of copyright is predicated upon the dual premises that the public benefits from the creative activities of authors, and that the copyright monopoly is a necessary condition to the full realization of such creative activities." 1 M. Nimmer, Nimmer on Copyright § 1.03[A] at 1-32 (1987).
Determining whether an aspect of an author's work is protected expression or unprotected idea is problematic where the alleged infringer borrows a nonliteral aspect of the copyrighted work. For example, copying the literal aspect of a novel—verbatim copying—is clearly an infringement of the author's monopoly in the expression. However, when an infringer borrows a nonliteral aspect of the novel, like the storyline, it becomes difficult to determine whether the infringer borrowed the idea of the novel or the author's expression of the idea.5 In the context of computer software, computer instructions are the literal aspect of the computer program. Nonliteral aspects include the "structure, sequence, and organization" of the computer instructions and the "look and feel" of the program as perceived by the user when the computer instructions are executed in the computer.

5. See, e.g., Nichols v. Universal Pictures Corp., 45 F.2d 119 (2d Cir. 1930). Judge L. Hand wrote:

[i]t is of course essential to any protection of literary property . . . that the right cannot be limited literally to the text, else a plagiarist would escape by immaterial variations. That has never been the law, but, as soon as literal appropriation ceases to be the test, the whole matter is necessarily at large. . . .

Id. at 121.

To demonstrate the difficulties of separating idea from expression consider Professor Nimmer's comparison of "Romeo and Juliet" to "Abie's Irish Rose" and "West Side Story":

Assuming Shakespeare's works were not in the public domain it is clear that no one could copy his drama concerning the star-crossed lovers of Verona merely by paraphrasing the dialogue with new (and no doubt less inspired) phrases. It is likewise clear that anyone could borrow the "idea" of a romance between members of two hostile families. Countless stories and plays have been based upon this abstract idea. Perhaps the most famous of these in more recent times was "Abie's Irish Rose." No more of "Romeo and Juliet" is contained in "Abie's Irish Rose" than the above idea. . . . Contrast this with the musical play and motion picture, "West Side Story." Since this involved a romance between members of two warring juvenile gangs in contemporary New York City, at first glance it might appear that this would no more constitute an infringement of "Romeo and Juliet" than does "Abie's Irish Rose." Certainly the dialogue and setting, and even much of the characterization, story line and action are far removed from the Shakespeare play. . . . but the essential sequence of events, as well as the interplay of the characters are straight out of "Romeo and Juliet."

3 NIMMER, supra note 3, § 13.03[A] at 13-26 (footnotes omitted). Nimmer concludes that "West Side Story" borrows expression from "Romeo and Juliet." Id. at 13-26 to -27.
Idea/Expression Continuum: literal vs. nonliteral aspects.

<table>
<thead>
<tr>
<th>idea</th>
<th>expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>function</td>
<td>look structure,</td>
</tr>
<tr>
<td>or purpose</td>
<td>sequence, feel</td>
</tr>
<tr>
<td>of program</td>
<td>organization</td>
</tr>
<tr>
<td></td>
<td>object code</td>
</tr>
<tr>
<td></td>
<td>(machine-readable)</td>
</tr>
<tr>
<td></td>
<td>computer</td>
</tr>
<tr>
<td></td>
<td>instructions)</td>
</tr>
<tr>
<td></td>
<td>source code</td>
</tr>
<tr>
<td></td>
<td>(human-readable)</td>
</tr>
<tr>
<td></td>
<td>computer</td>
</tr>
<tr>
<td></td>
<td>instructions)</td>
</tr>
</tbody>
</table>

On a case by case basis, courts must determine whether a nonliteral aspect of a work is an unprotected idea or a protected expression. Drawing the line between idea and expression in nonliteral infringement cases is perhaps the most pressing issue in computer software litigation. First, as an evidentiary matter, it is difficult to know how to draw the line when proving copyright infringement of computer software. Courts must consider the extent to which expert testimony is relevant to this determination and what factors should be applied when separating idea from expression. Second, drawing the line too near the idea end of the spectrum leads to protection of works dedicated to the public domain. This converts the Copyright Act into a general misappropriation statute.

6. See infra note 46 and accompanying text.

7. Determining where the line should be drawn is equivalent to determining the scope of protection. "The scope of protection will be of great economic significance to the computer industry, possibly even more important than protection of the programming code itself." Note, "Look and Feel", supra note 2, at 412. The computer industry is more concerned with protection of the structure, sequence, and organization and the look and feel because these aspects of the computer program contain more creativity than the actual programming of code. See infra notes 52-61 and accompanying text. See also Note, "Look and Feel", supra note 2, at 412 n.89.

Illustrative of the ramifications of separating idea from expression, one Note opposed extending protection to look and feel in spite of the potential for creative expression. The Note concluded that there should be no protection because tests for proving copyright infringement do not effectively separate idea from expression. Id. at 415.

8. As one author noted, the substantial similarity test for proving copyright infringement is:

one of the most difficult and baffling [issues] in all of copyright law. Any test for substantial similarity invariably calls for judicial line drawing, and, as Judge Learned Hand once conceded, these lines seem arbitrarily placed no matter where drawn. Nevertheless, courts have attempted to fashion tests in order to provide guidance in detecting substantial similarity.


9. See infra notes 116-22 and accompanying text.
This Note addresses these issues in four parts. Part I traces the legislative history of copyright protection for computer software and the scope of copyrightable subject matter. Part II discusses the test for copyright infringement as applied to computer software and the admissibility of expert testimony to resolve the idea/expression dichotomy. Part III analyzes the factors used in separating idea from expression in nonliteral infringement cases, i.e., structure, sequence, and organization cases and look and feel cases. Part IV addresses the criticism that applying the current factors for separating idea from expression has lead to overprotection of computer software. Analytically, this Note resolves the evidentiary problem of proving copyright infringement in Parts II and III by suggesting a codified standard for separating idea from expression in nonliteral infringement cases. Part IV of the Note evaluates the public policy considerations and demonstrates that the proposed standard does not lead to overprotection.

I. History of Copyright Protection for Computer Software

A. Legislative History of the Copyright Act

As early as 1964, the United States Copyright Office announced that copyright protection for computer programs was possible under the Copyright Act of 1909.10 While copyright problems regarding computer programs were emerging under the 1909 Act, drafters of


In May 1964, the Copyright Office announced that it “had taken the position that copyright registration for computer programs is possible under the present law.” United States Copyright Office, Copyright Registration for Computer Programs, (1964), reprinted in 11 Bull. Copyright Soc’y U.S.A. 361 (1964). The Copyright Office made the following statement:

The registrability of computer programs involves two basic questions:

(1) Whether a program as such is the "writing of an author" and thus copyrightable, and (2) whether a reproduction of the program in a form actually used to operate or be "read" by a machine is a "copy" that can be accepted for copyright registration.

Both of these are doubtful questions. However, in accordance with its policy of resolving doubtful issues in favor of registration wherever possible, the Copyright Office will consider registration for a computer program as a "book" in Class A if:

(1) The elements of assembling, selecting, arranging, editing, and literary expression that went into the compilation of the program are sufficient to constitute original authorship.

(2) The program has been published, with the required copyright notice; that is, "copies" (i.e., reproductions of the program in a form perceptible or capable of being made perceptible to the human eye) bearing the notice have been distributed or made available to the public.

(3) The copies deposited for registration consist of or include reproductions in a language intelligible to human beings. If the only publication was in a form that cannot be perceived visually or read, that
the revised copyright law found it impossible to legislate the new and evolving field of computer technology.\textsuperscript{11} As a result, in 1974 Con-

more (e.g., a print-out of the entire program) would also have to be deposited.

\textsuperscript{11} Bull. Copyright Soc'y U.S.A. at 361.

11. In preliminary drafts of the revised copyright law the House Committee on the Judiciary attempted to draft legislation which would address some of the copyright problems created by computer programs. One section of the preliminary draft, which eventually evolved into 17 U.S.C. § 106, explicitly discusses exclusive rights as they applied to programming or operations of an information storage system. The exclusive rights section of the preliminary draft read:

§ 5. Exclusive Rights Comprised in Copyright...

(a) The right to copy or record. Copyright shall include the exclusive right to copy or record the work in any tangible medium of expression, now known or later developed, from which it can be visually or aurally perceived, either directly or with the aid of a machine or device. It shall include the right to reproduce the work in visual copies, to make or duplicate sound recordings of it, to make a translation, adaptation, or any other derivative work from it, and to reproduce it in any form in the programming or operation of an information storage and retrieval system.


Later drafts of the exclusive rights provision did not explicitly refer to computers or computer programs. \textit{See Copyright Law Revision, Part 5, 89th Cong., 1st Sess., 4 (1965), reprinted in \textit{CONTU, Final Report} at 82.} The Register of Copyrights explained that:

[I]t would be a mistake for the statute, in trying to deal with such a new and evolving field as that of computer technology, to include an explicit provision that could later turn out to be too broad or too narrow. A much better approach, we feel, is to state the general concepts of copyright in language, such as that in section 106(a) which would be general in terms and broad enough to allow for adjustment to future changes in patterns of reproduction and other uses of authors' works.

\textit{Copyright Law Revision, Part 6: 89th Cong., 1st Sess., 18 (1965), reprinted in \textit{CONTU Final Report} at 82-83.} \textit{See also H.R. Rep. No. 83, 90th Cong., 1st Sess. 24 (1967)} ("recognizing the profound impact that information storage and retrieval devices seem destined to have on authorship, communications, and human life itself, the committee is also aware of the danger of legislating prematurely in this area of exploding technology").

Finally in 1969, the Copyright Revision Bill was amended to expressly protect computer programs; section 117 read:

Notwithstanding the provisions of sections 106 through 116, this title does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information or in conjunction with any similar device, machine or process, than those afforded to works under the law, whether title 17 or the common law or statutes of a State, in effect on December 31, 1970, as held applicable and construed by a court in an action brought under this title.

gress established the National Commission on New Technological Uses of Copyrighted Works (CONTU) to study and compile data on protection of copyrighted works of authorship in conjunction with computer uses. 12

Congress passed the Copyright Act of 1976 as Title 17 of the United States Code. 13 The new copyright act did not explicitly make computer programs copyrightable subject matter. The House of Representatives' report on the Act, however, stated that Congress intended computer programs to fall within the category of copyrightable "literary works." 14 Furthermore, section 117 of the Act verified the law's application to computer programs by defining the scope of


12. Immediately after the Senate passed the Copyright Revision Bill a second bill was brought before the Senate which established CONTU. Since it was unlikely that the House would take action on the Copyright Revision Bill, proponents of the bill felt that it was desirable to establish CONTU "to prepare for the resolution of the copyright issues which were arising from the rapid development of new technology." S. 3976, 93rd Cong., 2d Sess., 120 Cong. Rec. 30,516 (1974). CONTU was established in 1974 to:

[S]tudy and compile data on:

(1) the reproduction and use of copyrighted works of authorship

(A) in conjunction with automatic systems capable of storing, processing, retrieving, and transferring information, and

(B) by various forms of machine reproduction not including reproduction by or at the request of instructors for use in face-to-face teaching activities; and

(2) The creation of new works by the application or intervention of such automatic systems or machine reproduction.

(c) The commission shall make recommendations as to such changes in copyright law or procedures that may be necessary to assure for purposes access to copyrighted works, and to provide recognition of the rights of copyright owners.


14. Section 102(a) of the Act enumerates the works of authorship which qualify as proper subject matter for copyright.

(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, either directly or with the aid of a machine or device. Works of authorship include the following categories:

(1) literary works;

(2) musical works, including any accompanying words;

(3) dramatic works, including any accompanying music;

(4) pantomimes and choreographic works;

(5) pictorial, graphic, and sculptural works;

(6) motion pictures and other audiovisual works; and

(7) sound recordings.
copyright protection as it pertained to computer programs.\textsuperscript{15} The purpose of section 117 was merely to preserve the status quo, thereby leaving CONTU to resolve the computer related issues of authorship, distribution, and use with definitive copyright provisions.\textsuperscript{16}

\textsuperscript{15} Section 117 limits the scope of the exclusive rights defined in section 106 as such rights relate to computer and similar information systems. Section 117 reads: Notwithstanding the provisions of sections 106 through 116 and 118... this title... does not afford to the owner of copyright in a work any greater or lesser rights with respect to the use of the work in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information, or in conjunction with any similar device, machine, or process, than those afforded to works under the law, whether title 17... or the common law or statutes of a State, in effect on December 31, 1977, as held applicable and construed by a court in an action brought under this title. ...

\textsuperscript{16} The discussion and analysis from the House Report of section 117 of the Copyright Act states that:

As the program for general revision of the copyright law has evolved, it has become increasingly apparent that in one major area the problems are not sufficiently developed for a definitive legislative solution. This is the area of computer uses of copyrighted works: the use of a work “in conjunction with automatic systems capable of storing, processing, retrieving, or transferring information.” The Commission on New Technological Uses is, among other things, now engaged in making a thorough study of the emerging patterns in this field and it will, on the basis of its findings, recommend definitive copyright provisions to deal with the situation.

Since it would be premature to change existing law on computer uses at present, the purpose of section 117 is to preserve the status quo. It is intended neither to cut off any rights that may now exist, nor to create new rights that might be denied under the Act of 1909 or under common law principles currently applicable.

The provision deals only with the exclusive rights of a copyright owner with respect to computer uses, that is, the bundle of rights specified for other types of uses in section 106 and qualified in sections 107 through 116 and 118. With respect to the copyright-ability [sic] of computer programs, the ownership of copyrights in them, the term of protection, and the formal requirements of the remainder of the bill, the new statute would apply.

Under section 117, an action for infringement of a copyrighted work by means of a computer would necessarily be a federal action brought under the new title 17. The court, in deciding the scope of exclusive rights in the computer area, would first need to determine the applicable law, whether
CONTU recommended two statutory changes to the Copyright Act of 1976: 1) an addition to section 101 defining computer programs "to make it explicit that computer programs, to the extent that they embody an author's original creation, are proper subject matter of copyright;" and 2) a revision of section 117 "to ensure that rightful possessors of copies of computer programs may use or adapt these copies for their use." Congress responded to these recommended statutory changes by enacting the Computer Software Act of 1980, which amended the Copyright Act of 1976 to conform with CONTU's recommendations.

State statutory or common law or the Act of 1909. Having determined what law was applicable, its decision would depend upon its interpretation of what that law was on the point on the day before the effective date of the new statute.


17. CONTU, FINAL REPORT, supra note 11, at 1.
18. Id. The purpose of revising section 117 was to ensure that possessors of copyrighted computer programs would be allowed to use the programs freely to accomplish the tasks for which they are intended. CONTU recognized two problematic areas of copyright law as it applies to computer programs. First, the possessor should have the legal right to copy the program to the extent necessary to use the program.

This would include the right to load it into a computer and to prepare archival copies of it to guard against destruction or damage by mechanical or electrical failure. But this permission would not extend to other copies of the program. Thus, one could not, for example, make archival copies of a program and later sell some while retaining some for use. The sale of a copy of a program by a rightful possessor to another must be of all rights in the program, thus creating a new rightful possessor and destroying that status as regards the seller. This is in accord with the intent of that portion of the law which provides that owners of authorized copies of a copyrighted work may sell those copies without leave of the copyright proprietor.

Id. at 13 (footnote omitted). Second, the possessor should have the legal right to make changes to the program which are necessary to enable the use of the program for which it was both sold and purchased.

The conversion of a program from one higher-level language to another to facilitate use would fall within this right, as would the right to add features to the program that were not present at the time of rightful acquisition. These rights would necessarily be more private in nature than the right to load a program by copying it and could only be exercised so long as they did not harm the interest of the copyright proprietor. Unlike the exact copies authorized as described above, this right of adaptation could not be conveyed to others along with the licensed or owned program without the express authorization of the owner of the copyright in the original work. Preparation of adaptations could not, of course, deprive the original proprietor of copyright in the underlying work. The adaptor could not vend the adapted program, under the proposed revision of the new law, nor could it be sold as the original without the author's permission.

Id. (footnotes omitted).

19. Act of December 12, 1980 (Computer Software Act), Pub. L. No. 96-517, § 10(a), 94 Stat. 3028 (codified at 17 U.S.C. § 101 (Supp. 1982)). Section 101 defined "computer program" as "a set of statements or instructions to be used directly
Prior to the Computer Software Act of 1980, only one case squarely addressed the issue of copyright infringement of computer programs. The court properly ruled that the 1976 version of section 117 did not alter copyright protection as applied to computer uses. It further ruled that common law and the 1909 Act were controlling. The court concluded that copyright protection would not extend beyond the source code (human-readable computer instructions) to object code (machine language instructions). Dicta from

or indirectly in a computer in order to bring about a certain result." 17 U.S.C. § 101 (1982). The revised section 117 reads:

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


20. Data Cash Sys., Inc. v. JSA Group, Inc., 480 F. Supp. 1063 (N.D. Ill. 1979). In Data Cash, the defendant marketed a chess computer program stored in ROM (read only memory). The plaintiff had copyright registration on the source code, which generated the object code program in ROM, but not on the object code itself. There was no genuine issue of material fact and both parties moved for summary judgment. Id. at 1066. Since the object code stored in the two ROMs was identical, the court assumed direct copying. Id. The court found, however, that object code is not copyrightable subject matter. See id. at 1068.

21. The court noted that the legislative history of section 117 indicates an intent "to preserve the status quo" and that it was not intended to "cut off any rights . . . or to create new rights . . . ." Hence, the court concluded that it was to apply the law as it stood prior to the 1976 Act, i.e., the court was to apply common law and the 1909 Act. Id. at 1067. Alternatively, "[e]ven if the 1976 Act did apply, copying of the ROM would not be actionable," since the ROM would not constitute a "copy" under the definition in section 101. Id. at 1066-67, n.4. While the definition of "copy" encompasses works which may be perceived "with the aid of a machine or device," the court believed that only flow chart, source, and assembly phases but not object/ROM phases would constitute a copy. The court reasoned that ROM is a mechanical device, and "[m]echanical devices which cannot qualify as pictorial, graphic or sculptural works are not writings and may not obtain copyright protection." Id.

22. The court determined that "[b]oth at common law and under the 1909 Act, a 'copy' must be in a form which others can see and read." Id. at 1068. Since ROM is not in such a form, it cannot be a copy under common law or the 1909 Act and there is no infringement. Id. at 1068-69. The court of appeals affirmed the summary judg-
an earlier case, however, indicated that copyright protection would extend beyond source code to the structure, sequence, and organization of a program.23

Passage of the Computer Software Act has prompted a good deal of litigation concerning the protection of various aspects of computer programs. Courts have extended copyright protection to: source code,24 object code,25 database information,26 audiovisual

\[\text{See Data Cash Sys., Inc. v. JS&A Group, Inc., 628 F.2d 1038, 1044 (7th Cir. 1980).}\]

23. Synercom Technology, Inc. v. University Computing Co., 462 F. Supp. 1003 (N.D. Tex. 1978). In Synercom, the court stated in dicta that it is clearly an infringement to copy a program from one language to another. Likewise, it would probably also be an infringement to copy the flow-chart or pseudo-code, i.e., the structure of the program, in creating the source code. \[\text{Id. at 1013, n.5.}\] The court also stated:

It has long been the law that 

\[\ldots [i]n cases of literary or artistic works, and works of similar character, in which the form, arrangement or combination of ideas represents the product of labor and skilled effort separate and apart from that entailed in the development of the intellectual conception involved, that in such a situation, the medium of expression is entitled to protection by copyright against its adoption by another in similar form, arrangement, and combination.\]

\[\text{Id. at 1012-13 (citing Long v. Jordan, 29 F. Supp. 287, 288 (N.D. Cal. 1939)).}\]

24. \[\text{See, e.g., CMS Software Design Sys., Inc. v. Info Designs, Inc., 785 F.2d 1246, 1249 (5th Cir. 1986) (reversed and remanded trial court’s grant of Rule 41(b) involuntary dismissal motion where defendant marketed accounting programs with 80-90% of its source code being identical to plaintiff’s programs); Bly v. Banbury Books, Inc., 638 F. Supp. 983, 985 (E.D. Pa. 1986) (infringement found when licensee made an unauthorized copy of plaintiff’s program by loading it from plaintiff’s master disk into main memory after license was cancelled); E.F. Johnson Co. v. Uniden Corp. of America, 623 F. Supp. 1485, 1496 (D. Minn. 1985) (preliminary injunction granted, when expert testimony revealed that 38 out of 44 subroutines found in plaintiff’s program were duplicated in defendant’s program); Micro-Sparc, Inc. v. Amtype Corp., 592 F. Supp. 33, 34 (D. Mass. 1984) (infringement found where defendant typed copyrighted programs onto a masterdisk and then sold copies of the masterdisk); Videotronics, Inc. v. Bend Electronics, 586 F. Supp. 478, 480 (D. Nev. 1984) (court ordered preliminary injunction against defendant’s sales and marketing of program written by plaintiff).}\]

25. \[\text{See, e.g., M. Kramer Mfg. Co., Inc. v. Andrews, 783 F.2d 421, 446 (4th Cir. 1986) (infringement found where defendant’s object code was virtually identical to plaintiff’s object code and included a hidden legend that would appear on the display only when certain buttons were pushed in an abnormal sequence); Apple Computer, Inc. v. Formula Int’l, Inc., 725 F.2d 521, 523-25 (9th Cir. 1984) (preliminary injunction ordered where defendant manufactured and sold ROM chips storing the plaintiff’s operating system); Apple Computer, Inc. v. Franklin Computer Corp., 714 F.2d 1240, 1253 (3d Cir. 1983), \text{cert denied,} 464 U.S. 1033 (1984) (reversing denial of preliminary injunction where defendant copied several ROM programs written by plaintiff and plaintiff was able to show that defendant could have created its own programs to accomplish the same task); Williams Elec., Inc. v. Artic Int’l, Inc., 685 F.2d 870, 876-77 (3rd Cir. 1982) (injunction ordered to prevent defendant from marketing a circuit board “kit” using a program “virtually identical” to plaintiff’s); Atari, Inc. v. JS&A Group, Inc., 597 F. Supp. 5, 7 (N.D. Ill. 1983) (preliminary injunction ordered where defendant manufactured a device which would make copies of games written\]
output,\textsuperscript{27} program structure, sequence, and organization,\textsuperscript{28} and more recently, to the look and feel of the program.\textsuperscript{29} An unlawful

by plaintiff); Tandy Corp. v. Personal Micro Computers, Inc., 524 F. Supp. 171, 173 (N.D. Cal. 1981) (denying defendant's motion to dismiss, ruling that copying of silicon chip containing computer instructions is infringement). \textit{Contra Data Cash}, 628 F.2d 1038, 1043-44 (affirming summary judgment motion for defendant, determining that copying program on ROM chip was not infringement where there was no copyright notice on chip, gameboard, packaging, or accompanying instructions).

26. \textit{See}, \textit{e.g.}, Koontz v. Jaffarian, 787 F.2d 906, 909 (4th Cir. 1986) (trial court's finding of infringement upheld where defendant used plaintiff's data compilation to create a data base substantially similar to plaintiff's); Rand McNally & Co. v. Fleet Management Sys., Inc., 634 F. Supp. 604, 607-08 (N.D. Ill. 1986) (infringement found where defendant created a data base by copying data from mileage charts compiled by plaintiff).

27. Audiovisual output is analyzed differently than the computer uses formerly discussed, in that audiovisual output is deemed appropriate copyright subject matter under 17 U.S.C. § 102 (a)(6), "motion pictures and other audiovisual works," as opposed to 17 U.S.C. § 102 (a)(1), "literary works". Courts ruling that copyright protection extends to audiovisual output have considered the issue in relation to protection of video game displays. \textit{See}, \textit{e.g.}, \textit{M. Kramer Mfg. Co., Inc.}, 783 F.2d at 434-42; Midway Mfg. Co. v. Artic Int'l), Inc., 704 F.2d 1009, 1011-12 (7th Cir. 1983); \textit{Williams Electronics, Inc.}, 685 F.2d at 873-75; Universal City Studios, Inc. v. Nintendo Co., 615 F. Supp. 838, 856-57 (S.D.N.Y. 1985) \textit{aff'd}, 797 F.2d 70 (2d Cir. 1986).

28. \textit{See}, \textit{e.g.}, Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222 (3rd Cir. 1986), \textit{cert. denied}, 107 S. Ct. 877 (1987) (infringement found due to substantial similarity in file structure and management, screen output generation, and five subroutines' structure); Williams v. Arndt, 626 F. Supp. 571, 579 (D. Mass. 1985) (infringement found where defendant developed a computer program based on a commodities market trading system described in plaintiff's book); SAS Institute, Inc. v. S & H Computer Sys., Inc., 605 F. Supp. 816 (M.D. Tenn. 1985) (infringement found where structure of defendant's program was substantially similar to plaintiff's program). \textit{See also} Q-Co Indus., Inc. v. Hoffman, 625 F. Supp. 608, 615-16 (S.D.N.Y. 1985) (although the court recognized the relevance of structure, sequence, and organization in considering copying, it found the idea of plaintiff's program, rather than the expression of an idea was copied, and thus found no infringement); Synercom Technology, Inc. v. University Computing Co., 462 F. Supp. 1003, 1012-14 (N.D. Tex. 1978) (finding plaintiff's arrangement of data in input formats is expression). \textit{Contra} Plains Cotton Co-op v. Goodpasture Computer Serv., 807 F.2d 1256, 1262 (5th Cir. 1987) (finding structure, sequence, and organization not protectable).


Look and feel cases are distinguishable from audiovisual cases. \textit{See supra} note 27. In audiovisual cases the question is whether protection extends to the screens when the author has copyrights on the screens. In look and feel cases the question is
appropriation of any one of these aspects of a computer program may give rise to a copyright infringement action.

II. THE TEST FOR PROVING COPYRIGHT INFRINGEMENT OF COMPUTER SOFTWARE

In order to prevail in an action for copyright infringement, the plaintiff must prove his ownership of the copyright and a violation of an exclusive right such as copying by the defendant. Since a plaintiff is generally unable to produce evidence of actual copying, "copying may be proved inferentially by showing that the defendant had whether protection extends to the screens and other aspects of the computer program perceived by the user when the author has a copyright on the program.


[T]he owner . . . has the exclusive rights to do and to authorize any of the following:

1. to reproduce the copyrighted work in copies or phonorecords;
2. to prepare derivative works based upon the copyrighted work;
3. to distribute copies or phonorecords of the copyrighted work to the public by sale or other transfer of ownership, or by rental, lease, or lending;
4. in the case of literary, musical, dramatic, and choreographic works, pantomimes, and motion pictures and other audiovisual works, to perform the copyrighted work publicly; and
5. in the case of literary, musical, or dramatic, and choreographic works, pantomines, and pictorial, graphic, or sculptural works, including the individual images of a motion picture or other audiovisual work, to display the copyrighted work publicly.


Professor Nimmer describes ownership as consisting of five elements:

1. Originality in the author, (2) copyrightability of the subject matter, (3) citizenship status of the author such as to permit a claim of copyright, (4) compliance with applicable statutory formalities, and (5) (if the plaintiff is not the author) a transfer of rights or other relationship between the author and the plaintiff so as to constitute the plaintiff the valid copyright claimant.

access to the allegedly infringed copyrighted work and that the allegedly infringing work is substantially similar to the copyrighted work.”

107 S. Ct. 962 (1987). Nonetheless, the first two elements of ownership are quite often contested by the other party.

The originality requirement obtains authority from section 102(a) of the Copyright Act. Copyright protection extends to “original works of authorship fixed in any tangible medium of expression.” 17 U.S.C. § 102(a) (1982). See West Pub. Co. v. Mead Data Cent., Inc., 799 F.2d 1219, 1225 (8th Cir. 1986); National Football League v. McBee & Bruno’s, Inc., 792 F.2d 726, 731 (8th Cir. 1986); Toro Co. v. R & R Products Co., 787 F.2d 1208, 1212 (8th Cir. 1986); Hutchinson Telephone Co v. Fronteer Directory Co. of Minn., Inc., 770 F.2d 128, 131 (8th Cir. 1985); Donald v. Uarco Business Forms, 478 F.2d 764, 765 (8th Cir. 1973). “The legislative history states that the phrase ‘original works of authorship’ was intended to codify without change the concept of ‘originality’ applied by the courts under the prior copyright statute.” Hutchinson Telephone Co., 770 F.2d at 131 (citing, H.R. REP. No. 1476, 94th Cong., 2d Sess., reprinted in 1976 U.S. CODE CONG. & ADMIN. NEWS at 5659, 5664).

See also Toro Co., 787 F.2d at 1211-12.

“The standard for ‘originality’ is minimal. It is not necessary that the work be novel or unique, but only that the work have its origin with the author — that it be independently created.” West Pub. Co., 799 F.2d at 1223. See also Toro Co., 787 F.2d at 1212; Hutchinson Telephone Co., 770 F.2d at 131; Donald, 478 F.2d at 765. Originality requires only that the author invest some “creative intellectual or aesthetic labor.” Goldstein v. California, 412 U.S. 546, 561 (1973); West Pub. Co., 799 F.2d at 1223. Even a very slight degree of labor may be sufficient so long as it distinguishes authorship and is not merely a trivial variation from other works. See id.; Toro Co., 787 F.2d at 1212; Hutchinson Telephone Co., 770 F.2d at 131; Donald, 478 F.2d at 766.

The subject-matter requirement also derives its authority from section 102(a) of the Copyright Act. The section lists the works of authorship which are copyrightable. 17 U.S.C. § 102(a) (1982). Many computer applications are copyrightable. Courts have extended copyright protection to source code, object code, data base information, audiovisual output, structure, sequence, and organization, and look and feel. See supra notes 24-29 and accompanying text. To date, only one Minnesota court has considered the scope of copyright protection of computer software and ruled in favor of extending protection to computer source code. See E.F. Johnson Co., 623 F. Supp. at 1496.


Generally, copyright infringement consists of two elements: ownership and copying. Obtaining direct evidence of copying is problematic. First, copying is usually intentional, and therefore, concealed, particularly from the owner of the copyrighted work. Second, independent creation of the exact same work is not infringement. Courts will avoid these problems by allowing circumstantial evidence of copying. Evidence that the defendant had access to the copyrighted work and that the allegedly infringing work is substantially similar to the copyrighted work provides the logical inference that the defendant copied the plaintiff’s work. See, e.g., E.F. Johnson Co., 623 F. Supp. at 1492; 3 NIMMER, supra note 30, § 13.01[B].

Access has been defined as actual viewing and knowledge of the copyrighted
“The test of substantial similarity in the Eighth Circuit is ‘whether the work is recognizable by an ordinary observer as having been taken from the copyrighted source’.”32 The “ordinary observer” test supports a finding of substantial similarity where, without the aid or suggestion of expert testimony, the “spontaneous and immediate” reaction of the public is that the works are substantially similar.33

Courts have analytically split the “ordinary observer” test into two steps when comparing sophisticated works beyond the understanding of the ordinary observer. The two-step approach was initially set forth in *Arnstein v. Porter*34 where the court held that under the first step, expert testimony is admissible to define the similarities between work by the infringer. See *Bradbury v. Columbia Broadcasting Sys., Inc.*, 287 F.2d 478, 479 (9th Cir. 1961), *cert. dismissed*, 368 U.S. 801 (1961). Proof of actual viewing, however, may be as difficult to obtain as proof of actual copying. See 3 *Nimmer*, *supra* note 30, § 13.02[A]. Often an opportunity to view will be sufficient to create an inference of access. Smith v. Little, Brown & Co., 245 F. Supp. 451, 458 (S.D.N.Y. 1965), *aff’d*, 360 F.2d 928 (2d Cir. 1966). Other courts have accepted a reasonable opportunity to view as direct evidence of access. See *Kamar Int’l, Inc. v. Russ Berrie & Co.*, 657 F.2d 1059, 1062 (9th Cir. 1981); *Ferguson v. National Broadcasting Co., Inc.*, 584 F.2d 111, 113 (5th Cir. 1978); Sid & Marty Krofft Television Prods., Inc. v. McDonald’s Corp., 562 F.2d 1157, 1162 (9th Cir. 1977). In the context of computer software, a defendant’s awareness of the existence of a widely distributed copyrighted work constituted proof of access. *Midway Mfg.*, 546 F. Supp. at 145.


33. Harold Lloyd Corp., 65 F.2d at 18 (9th Cir. 1933). Under the ordinary observer test, substantial similarity is found when the allegedly infringing work has captured the “total concept and feel” of the copyrighted work. See *Roth Greeting Cards v. United Card Co.*, 429 F.2d 1106, 1110 (9th Cir. 1970); *E.F. Johnson Co.*, 623 F. Supp. 1485, 1492-93 (citing *Atari, Inc. v. North America Philips Consumer Elec. Corp.*, 672 F.2d 607, 614 (7th Cir. 1982)). An expert is not better qualified to judge the “concept and feel” of a work than is an ordinary observer. Moreover, expert testimony would only lead to “the classic difficulty of not being able to see the forest for the trees.” Frankel v. Irwin, 34 F.2d 142, 144 (S.D.N.Y. 1918).

34. 154 F.2d 464 (2d Cir. 1946). In *Arnstein*, the court reasoned that in applying the ordinary observer test, “it is important to avoid confusing two separate elements essential to a plaintiff’s case in such a suit: (a) that the defendant copied from the
the two works.\textsuperscript{35} Provided the similarities are sufficiently substantial to infer copying, the second step requires that the ordinary observer determine whether there was an improper appropriation of the copyrighted work.\textsuperscript{36}

In \textit{Sid & Marty Krofft v. McDonald's Corp.},\textsuperscript{37} the 9th Circuit interpreted \textit{Arnstein} as admitting expert testimony to evidence only similarities in idea. If the idea is sufficiently similar to infer copying, the ordinary observer is given the task of detecting similarities in expression.\textsuperscript{38} Commentators following the \textit{Krofft} analysis of \textit{Arnstein} have argued that the two-step approach is not workable when applied to computer software.\textsuperscript{39} The criticism is that it is unreasonable to ex-

\begin{footnotesize}

35. The \textit{Arnstein} court stated that "[i]f there is evidence of access and similarities exist, then the trier of the facts must determine whether the similarities are sufficient to prove copying. On this issue, analysis ("dissection") is relevant, and the testimony of experts may be received to aid the trier of the facts." \textit{Arnstein}, 154 F.2d at 468.

36. \textit{Id.}

37. 562 F.2d 1157 (9th Cir. 1977).

38. \textit{See Krofft}, 562 F.2d at 1165. \textit{Krofft} interpreted \textit{Arnstein} as "alluding to the idea-expression dichotomy..." \textit{Id.} Under the \textit{Krofft} interpretation, "copying" refers merely to the permissible copying of ideas. The second step, improper appropriation, refers to the unlawful copying of expressions. \textit{Id.}


\end{footnotesize}
pect that an ordinary observer can detect similarities in the expression of a computer program. While it is true that the ordinary observer cannot dissect a computer program and define the similarities, this criticism is based on a misinterpretation of the Arnstein test. The Arnstein two-step approach admits expert testimony to evidence similarities in both idea and expression.


40. One author argues that the current methods of proving substantial similarity are not workable, stating:

Though courts now generally agree that copyright protection extends to most forms of software, they lack a workable standard for determining when infringement of copyrights in computer programs has occurred . . . . The few courts that [have decided] whether a program constituted an illicit copy of an original computer program, however, have either ignored the substantial similarity test altogether, or stated that they were applying the test and then ignored its standards.

Note, Substantial Similarity, supra note 39, at 1264-65. The author contends that under both the traditional ordinary observer test and the Arnstein-Krofft two-step approach, the ordinary observer is required to detect similarities in expression. Id. at 1280. This is impossible when the subject matter is computer software, unless there is expert evidence and detailed dissection of the work. Id. at 1280-84. An ordinary observer test is therefore not workable. Id. at 1285-88. See also Note, Unified Theory of Copyright Infringement, supra note 39, at 468 (1982) (the ordinary observer test is not a good standard when the copyrighted work is a sophisticated technology).

41. Professor Nimmer argues that the Krofft interpretation of Arnstein is incorrect. He points out that three of four cases cited in Arnstein to illustrate copying “are examples of similarity of expression not merely of idea.” 3 NIMMER, supra note 30, § 13.03[E] n.121.7. Under Arnstein, the first step “probably calls for a determination of whether there is substantial similarity of expression (not merely of idea), but this is to be made regardless of the protectibility of such expression, while the [second step] of substantial similarity is confined to that expression which is protectible by the plaintiff.” Id. at § 13.03[E].

Moreover, the Krofft reading of Arnstein is virtually absurd. If copying refers merely to evidence of using the same idea, the first step of the test is useless. Evidence of using the same idea employed by the plaintiff does not tend to prove copying of the plaintiff’s copyrighted expression.

One commentator had the following thoughts on the Krofft interpretation of Arnstein.

Krofft interprets Arnstein’s two-step analysis to be the same idea/expression, extrinsic-intrinsic test articulated in Krofft itself. Assuming arguendo that this questionable interpretation of Arnstein is correct, the Krofft court’s analysis suffers from two major weaknesses.

First, since the Copyright Act does not protect ideas, the entire Krofft extrinsic test is irrelevant to a copyright infringement action. As a practical matter, any infringement suit in which the ideas are so disparate as to not be substantially similar is a likely candidate for summary judgment.
Under the *Arnstein* approach, the only task left to the ordinary observer is to resolve the second step in the two-step test, determining whether there was an "improper appropriation".\(^{42}\) It should be recognized since copyrightable subject matter is the particular expression of an idea, not the idea in its purest form,\(^{43}\) an appropriation is "improper" only when expression is borrowed. Furthermore, proof of infringement is predicated upon a finding of substantial similarity between the two works.\(^{44}\) It follows that in order to find an "improper appropriation," the ordinary observer must determine that a substantial amount of expression was borrowed. The expert witness does not impinge on this determination by dissecting the program, defining similarities in idea and expression, and elaborating on technical reasons for the similarities. Instead, the expert testimony merely provides the technical background to determine if a substantial amount of expression was borrowed.

Restating the *Arnstein* two-step approach in light of this understanding of "improper appropriation", it is apparent that the two-step approach is workable. First, expert testimony is admitted to evidence all similarities in the two works regardless of whether the similarities are of idea or expression. Further, the expert may supply the

---

*Krofft's second problem lies in the restriction placed on the intrinsic test.*

The *Krofft* court held that dissection and expert testimony are irrelevant to the determination of substantial similarity in the expression of ideas, and that the fact finder should consider only whether the accused work has captured the "total concept and feel" of the protected work. However, the "total concept and feel" test identifies only similarity between two ideas, which may not be copyrighted, not between two expressions of an idea, which may be copyrighted. Absent dissection and analysis, sanctioned in *Arnstein*, *Krofft's* "total concept and feel" test is essentially indistinguishable from the older, unworkable audience test.


The trend is to increase the use of expert testimony when separating idea from expression. *Whelan*, 797 F.2d at 1233; *Broderbund*, 648 F. Supp. at 1136; *E.F. Johnson*, 623 F. Supp. at 1493. See also *Fed R. Evid.* 702 (expert testimony should be admitted when it can help the trier of fact). Expert testimony is necessary since:

[without guidance, an ordinary [observer] may be unable to detect well-disguised or highly modified programs. Using only the ordinary observer audience test and their progeny [as in *Krofft*], the trier of fact may be easily misled by unimportant factors when confronted with two disparate-looking programs. Expert testimony in software copyright infringement cases may be indispensable, and mechanical application of any infringement test which prohibits such testimony may preclude a correct disposition.


\(^{42}\) *Arnstein*, 154 F.2d at 468.


\(^{44}\) See supra note 31 and accompanying text.
ordinary observer with any other technical support needed to separate similarities in idea from similarities in expression. Second, the ordinary observer can then determine which similarities are similarities in expression and whether the similarities in expression constitute a substantial amount of the copyrighted work as a whole.\footnote{For example, if expert testimony revealed that the two works were similar in purpose, the ordinary observer may determine that the works are similar in idea only and that the appropriation was proper. On the other hand, if expert testimony revealed that the two works contained 200 lines of identical code and that there was no technical reason for using identical source code, the ordinary observer may determine that the similarity in code was protectable expression. If the use of the code constituted a substantial amount of the program, there would be an improper appropriation.}

The difficult issue for the ordinary observer is in separating the similarities into idea and expression. There are no concrete rules. In the words of Judge Learned Hand, the author of various leading copyright opinions, "[o]bviously, no principle can be stated as to when an imitator has gone beyond the 'idea,' and has borrowed its 'expression.' Decisions must therefore inevitably be ad hoc."\footnote{Peter Pan Fabrics, Inc. v. Martin Weiner Corp., 274 F.2d 487, 489 (3d Cir. 1960). \textit{See also} Nichols v. Universal Pictures Co., 45 F.2d 119, 121 (2d Cir. 1930).}

Nonetheless, the factors for separating idea from expression in nonliteral infringement cases are illustrative of how this determination is made by the ordinary observer.

III. THE STANDARD FOR SEPARATING IDEA FROM EXPRESSION IN NONLITERAL INFRINGEMENT CASES


More recently, the issue has been raised in look and feel cases including \textit{Broderbund Software, Inc. v. Unison World, Inc.}\footnote{648 F. Supp. 1127 (N.D. Cal. 1986).} and \textit{Digital Communications Associates, Inc. v. Softklone Distributing Corp.}\footnote{659 F. Supp. 449 (N.D. Ga. 1987).}

A. Expression May Exist in Nonliteral Aspects of Computer Software

Creation of a computer program may be characterized in five parts: 1) defining the purpose and function of the program; 2) designing the data structure; 3) designing the modular structure;
4) designing the user interface; and 5) coding the program design from parts two, three, and four into source code.\textsuperscript{52} While the literal aspect of a program, the source code, is unequivocally protected with a copyright on the program, the most creative expression resides in the nonliteral aspects of designing the program.

To date, no court has squarely addressed the issue of whether part two, the data structure, is protectable as a nonliteral aspect of the copyrighted program.\textsuperscript{53} The data structure is the logical relationship of information stored by the program. As the program is executed, it uses information that resides in a memory storage device. While the physical location of the information is irrelevant, operation of the program depends on the logical relationship of the information as defined by the programmer. The logical data structure is often arranged as a matter of programming style.

Three courts have considered whether part three—the modular structure, otherwise known as structure, sequence, and organization—is protectable as a nonliteral aspect of the copyrighted program.\textsuperscript{54} A computer program is composed of related procedures which operate together to bring about a desired result. The program structure, sequence, and organization is the way in which the procedures relate. Designing the structure, sequence, and organization “takes place in several steps, moving from general to specific.”\textsuperscript{55} A few very general tasks known as modules or subroutines are created. Each procedural module is broken down into successively smaller and more detailed tasks until each detailed module performs essentially one simple task.\textsuperscript{56} The structure, sequence, and organiza-

\textsuperscript{52} The author studied undergraduate computer science and has several years experience as a computer programmer and consultant. The following section, describing the different aspects of a computer program, is generally supported by the author’s personal knowledge of computer programming.

\textsuperscript{53} However, there was some evidence of copying of the plaintiff’s data structure in Whelan. 797 F.2d at 1228.

\textsuperscript{54} See supra notes 47-49 and accompanying text.

\textsuperscript{55} Whelan, 797 F.2d at 1229. See generally D. Bender, Computer Law Software Protection § 2.06[3] (1985) (discussing the creation of a computer program).

\textsuperscript{56} For example, if the programmer were to design a program which would allow the user to: 1) add, edit, and delete mailing labels in a computerized database; and 2) print-out all or some of the mailing labels in alphabetical order by last name, the programmer could adopt the following general modules.

\textbf{Step 1:} Determine whether user requests: 1) modification of the database (go to step 2), 2) output of mailing labels (go to step 3), or 3) end program (call this MAIN-MENU MODULE).

\textbf{Step 2:} Add mailing label to database or delete mailing label from database or edit mailing label in database. —return to step 1— (call this MODIFY-FILE MODULE).

\textbf{Step 3:} Create a list of mailing labels and print the list. —return to step 1— (call this PRINT-LABELS MODULE).

Within each one of these general modules, the programmer would develop more detailed modules. For example, within the PRINT-LABELS MODULE, the program-
tion may contain creative expression since decisions regarding the breakdown and organization of procedures are based on the preferences and programming style of the programmer. As computer programs become more complex, there is more opportunity for creative arrangement of the procedures.\textsuperscript{57}

Designing part four—the user interface, otherwise known as look and feel—is potentially the most creative aspect of the program. Two courts have considered whether look and feel is protectable as a nonliteral aspect of the copyrighted program.\textsuperscript{58} The look and feel is

The alphabetizing function is a simple task that would not require further breakdown. A common method for alphabetizing is the “bubble sort.” For example, if the task were to alphabetize the letters D, E, F, A, the following algorithm could be used:

\textbf{Bubble Sort Algorithm}

\begin{enumerate}
\item Compare the letter in first position with all letters to the right. Where letters are not alphabetical, swap their position and continue comparing.
\item Compare the letter in the second position with all letters to the right. Where letters are not alphabetical, swap their position and continue comparing.
\item Compare the letter in the third position with all letters to the right. Where letters are not alphabetical, swap their position and continue comparing.
\end{enumerate}

In operation, the letters would be manipulated as follows, beginning with: D E F A

\begin{enumerate}
\item Compare D and E; no change — D E F A
\item Compare D and F; no change — D E F A
\item Compare D and A; swap — A E F D
\end{enumerate}

\begin{enumerate}
\item Compare E and F; no change — A E F D
\item Compare E and D; swap — A D F E
\end{enumerate}

\begin{enumerate}
\item Compare F and E; swap — A D E F
\end{enumerate}

On the other hand, the programmer could choose to alphabetize the mailing labels as they are entered into the system as opposed to alphabetizing just prior to print-out. In that case, the alphabetizing module would fall out of the MODIFY-FILE MODULE rather than from the PRINT-LABELS MODULE. Clearly, at every turn a computer programmer must make decisions that effect the structure of the program. \textit{See Note, Defining the Scope of Copyright Protection for Computer Software, 38 STAN. L. REV. 497, 500-01 (1986) (discussing the breakdown of modules to smaller tasks).}

57. The mailing label program described supra note 56 is fairly simplistic. There are very few ways that the modules could be structured and organized. Designing this type of program may not involve any creative expression — the programmer would merely design the modular structure as his training dictates. In a more complex program, there may be several alternative modular designs allowing for creative expression.

58. \textit{See supra notes 50-51 and accompanying text.}
the program's "'design, presentation, and output as experienced by the user,' including the various observable attributes of the program such as commands, graphics, sounds, symbols, sequences, arrangement, and more general aesthetic qualities experienced by the user." The look and feel is often manifested in the design of video screens and command words which allow the computer and the user to interact. The virtually unlimited supply of video images and command words gives the programmer the widest possible creative discretion. This task is particularly creative when the programmer attempts to create a very "user friendly" system.

Not every program will contain protectable nonliteral expression. Nonliteral aspects of the program, however, may be the programmer's most creative work, and some amount of copyright protection must be available to protect these efforts.

B. Separating Idea from Expression in Structure, Sequence, and Organization Cases

SAS, Q-Co, and Whelan are nonliteral infringement cases where the copyright owner claimed that the alleged infringer borrowed the structure, sequence, and organization of the copyrighted work. Using the copyrighted program's structure, sequence, and organization, the infringers were able to write new source code which operated on a different computer system. In each of the three

59. Note, "Look and Feel", supra note 2, at 416.
60. See, e.g., Softkline, 659 F. Supp. at 452; Broderbund, 648 F. Supp. at 1137.
61. The Copyright Act provides protection for "original works of authorship fixed in a tangible medium of expression." 17 U.S.C. § 102 (1982). However, the protection does not extend to ideas. Id. This Note will conclude that protection of expression will extend to the creative efforts by the programmer and that the unprotected idea is the functional aspect of the program. Further, creative efforts necessary to the function of the program are merged with the function/idea and are also unprotectable.

The purpose of this section is merely to demonstrate that nonliteral aspects of the program are creative efforts that might be protected if not too closely linked to the function of the program.

64. 797 F.2d 1222 (3d Cir. 1986).
65. In SAS, the plaintiff owned a copyrighted statistical analysis program known by its registered trademark as "SAS". SAS was designed to be run on IBM and IBM compatible computers. The alleged infringing program, known as "INDAS" was developed by S & H Computer Systems to be a conversion of SAS, that would operate on a VAX computer. SAS, 605 F. Supp. at 818.

In Q-Co, the plaintiff owned a copyrighted text-prompting program known as "VPS-500". A text prompting program causes the computer to scroll large characters on a television screen to permit the speaker to read a prepared text by looking directly ahead at the screen, thereby avoiding the appearance of reading. VPS-500 was designed to be run on an ATARI 800-XL computer. Here, as in SAS, the defend-
cases the court found evidence of access and substantial similarity.\(^{66}\) The \(Q-Co\) court, however, held that the defendant's program did not infringe.\(^ {67}\)

In \(Q-Co\), the defendant copied four modules from the plaintiff's text-prompting program.\(^ {68}\) The court held that there was no infringement since the plaintiff's modular structure was not copyrightable expression.\(^ {69}\) Every text-prompting program would use the same four modules embodying the same functions. The modular structure employed was, therefore, inherent given the idea, and no other expression could have been used.\(^ {70}\) Nonetheless, the court did merely convert the plaintiff's program so that it would run on a different computer system. The defendant's program was known as "CPC-1000", and it operated on an IBM PC. While the ATARI's graphics chip made the ATARI better suited as a prompter, the IBM had greater memory which allowed it to process more text. \(Q-Co\), 625 F. Supp. at 610-13.

In \(Whelan\), the plaintiff owned a copyrighted dental business program known as "Dentalab". The program was designed to perform bookkeeping and administrative tasks. Dentalab was designed to run on an IBM Series One machine. The alleged infringing program known as "Dentcom" was developed by the defendant as a conversion of Dentalab that would run on smaller computer systems. Originally, Dentalab was designed by the plaintiff for use in the defendant's lab. The defendant later became a sales representative of the Dentalab program. As the defendant became more familiar with computers, he realized the sales potential of a Dentalab clone for smaller computers. He began converting Dentalab from EDL to BASIC in his free time, thereby creating the Dentacom PC program. \(Whelan\), 797 F.2d at 1225-26.

\(^{66}\) In \(SAS\), the court found that all of the S & H's programmers had access to SAS source code and documentation. The court further found substantial similarity since: 1) SAS Institute was able to cite forty-four incidents of copying; 2) evidence of copying SAS organization and structure pervaded the entire INDAS program; and 3) INDAS was developed in a fraction of the time required to develop SAS. In one incident of copying, INDAS source code contained a functionless source code which had accidently been left in the SAS source code after its intended function had been abandoned. \(SAS\), 605 F. Supp. at 828-30.

In \(Q-Co\), access was clearly established since the defendant was an ex-employee of the plaintiff and was instrumental in the design of VPS-500. The court found substantial similarity since CPC-1000 contained four modules which corresponded directly to four of twelve modules in VPS-500 and since the modules contained similar functions. \(Q-Co\), 625 F. Supp. at 611-12, 614.

In \(Whelan\), the district court established access based on the defendant's familiarity with Dentalab. The program was designed for the defendant's dental practice and he had access to the source code. The district court further found substantial similarity in the structure, sequence, and organization of the program since there was similarity in file structure, screen output, and five subroutines. \(Whelan\), 797 F.2d at 1242-46.

\(^{67}\) \(Q-Co\), 625 F. Supp. at 616.

\(^{68}\) \(Id.\) at 614.

\(^{69}\) "The evidence establishes that the modules in different languages were similar in the sense of 'ideas' rather than 'expressions' and as such have not been copied in the infringement sense." \(Id.\) at 615.

\(^{70}\) \(Id.\) at 616. Notice, however, that the court was only referring to the general
cite the earlier SAS case and recognized that under different circumstances, expression would extend to program structure, sequence, and organization.71

In SAS, the court held that expression extended to a program’s structure, sequence, and organization72 since developing program structure involves a virtually endless series of arbitrary choices.73 Thus, expression can be found wherever there is “room for variation” in creating a program’s structure, sequence, and organization. Where there are “only very limited ways” of creating the work, the structure, sequence, and organization is not protectable.74

Although the Q-Co court did not discuss the differing levels of sophistication between the SAS program and the Q-Co program, that distinction would reconcile the two cases. In SAS, there were several possible structure, sequence, and organization designs,75 and in Q-Co, the court apparently found that there was only one.76 SAS and Q-Co taken together indicate that copyrightable expression is more likely to exist in complex structure and organization designs as opposed to simplistic designs. A relatively simple structure, sequence, and organization can be designed in only a very limited number of ways. Such a design is dictated by necessity to the extent that the programmer will not design an inefficient program. In contrast, a complex program could be designed in a variety of ways, each of which might be equally efficient. Development of this type of program would be characterized by choice, not by necessity.

structure of the program, i.e., the four modules which were copied—title module, load module, prompting module, and large-characters module. Within each of these modules, however, is a more detailed structure which the court did not consider in its analysis. So while ideas may have vested in the general structure, unique expression may have existed in the more detailed structures of the individual modules.

71. Id.
73. Id. at 825. The court stated that:
[T]he author is faced with a virtually endless series of decisions as to how to carry out the assigned task. Beginning with a broad and general statement of the overall purpose of the program, the author must decide how to break the assigned task into smaller tasks, each of which must in turn be broken down into successively smaller and more detailed tasks. At the lowest levels the detailed tasks are then programmed in source code. At every level, the process [of developing a program’s structure, sequence, and organization] is characterized by choice often made arbitrarily, and only occasionally dictated by necessity.

Id.
74. Id.
75. See id. at 825 (“the court finds that to the extent that similarities between SAS and the S & H product have existed, they represent unnecessary intentional duplication of expression”).
76. Q-Co, 625 F. Supp. at 616.
The Whelan court took this proposition one step further with language which appears to apply generally to the idea/expression dichotomy and not just to structure, sequence, and organization cases. The court reasoned that "[w]here there are various means of achieving the desired purpose, then the particular means chosen is not necessary to the purpose; hence, there is expression, not idea."77 In general, structure, sequence, and organization cases separate idea from expression by determining whether the means chosen by the programmer were necessary or arbitrary.

Essentially, this is the same rule which was first promulgated in the landmark case, Baker v. Selden.78 In Baker, the court reasoned that "where the art it [the idea in the plaintiff's book] teaches cannot be used without employing the methods and diagrams used to illustrate the book, or such as are similar to them, such methods and diagrams are to be considered as necessary incidents to the art, and given therewith to the public . . . ."79

C. Separating Idea from Expression in Look and Feel Cases

Broderbund80 and Softklone81 are infringement cases in which the copyright owners claimed that the alleged infringers appropriated the video screens and command words of the copyrighted program, thereby capturing the total look and feel of the work. These cases follow the standard for separating idea from expression as set forth in the structure, sequence, and organization cases.82 There is an important factual difference, however, between structure, sequence, and organization cases and look and feel cases. A look and feel infringer can borrow a program's video screens and command words without ever accessing the actual computer program. Therefore, there is an issue as to whether look and feel is a nonliteral aspect of the computer program or a literal aspect of the video screens.83 If an alleged infringer borrows the literal aspect of a video screen rather than the nonliteral aspect of the program, copyright infringement

77. Whelan, 797 F.2d at 1236.
78. 101 U.S. 99 (1879).
79. Id. at 103.
82. See supra note 77 and accompanying text.
83. Although a particular computer program may generate certain video screens, a completely different program can generate the exact same screens. An alleged infringer may literally copy a video screen (verbatim copying) without copying the underlying program. Alternatively, an alleged infringer may copy the underlying program in order to generate similar video screens. In the first instance there is literal copying of the screens, and in the second instance, there is nonliteral copying of the program.
must be predicated on a copyrighted video screen, and cannot be predicated on a copyrighted program.

1. Structure, Sequence, and Organization Factors
   Applied in Look and Feel Cases

Generally, look and feel cases focus on the nexus between idea and expression. When idea and expression merge, it is impossible to protect expression without also creating a monopoly in the idea. As the *Softklone* court noted, *Baker* stands for the proposition that "ideas, as such, are not copyrightable and, as a corollary, necessary expression incident to an idea 'merge' with that idea and also are not copyrightable. . . ."


The issue in look and feel cases is whether the technical constraints of designing the video screens and command words cause the expression to merge with the idea. For example in *Broderbund*, Broderbund negotiated with Unison to have Unison develop a conversion of a program from Apple to IBM. When negotiations deteriorated, Unison created their own version of the program but retained Broderbund's menu screens and input formats. Since Broderbund was able to demonstrate to the court that a similar program expressed the same ideas in a different way, the court held that Unison had infringed upon Broderbund's copyrighted program. The expression in question was not a necessary incident to the underlying ideas.

Similarly, the *Softklone* court found that ForTec had infringed when it borrowed the copyrighted "status screen" from Microstuff's program. While the court found some aspects of the screen to be unprotectable ideas, it found that the command terms and the

---

84. "[I]f an idea is indistinguishable from its expression, that is, if the idea is 'merged' into its expression, the expression cannot be protected under the copyright laws." *Broderbund*, 648 F. Supp. at 1131.


87. *Id.* at 1131.

88. At trial, plaintiffs introduced another program that performed essentially the same function. However, this program generated very different screens. *Id.* at 1132. The court stated that "there do exist other, quite different ways of expressing the ideas embodied in [the copyrighted program]. The court rejects defendant's contention that the idea and expression of [the copyrighted program] are indistinguishable from one another." *Id.*

89. *Id.* at 1138.

90. *Softklone*, 659 F. Supp. at 465. Plaintiffs had developed a program called Crosstalk which enabled the user's computer to communicate with other computers. The program received widespread support because of its main menu called the "status screen". *Id.* at 452. The defendant's program served the same function and used a similar main menu. *Id.* at 453.

91. The court found that:
sequencing of commands were not a necessary incident to the idea.\textsuperscript{92} The choice of using two-letter commands and the sequencing was not technically constrained by programming needs.\textsuperscript{93} Since other choices where equally feasible, there was copyrightable expression.\textsuperscript{94}

The "merged idea" standard for separating idea from expression in \textit{Broderbund} and \textit{Softklone} is entirely consistent with the necessary/arbitrary distinction in \textit{Whelan}\textsuperscript{95} and other structure, sequence, and organization cases. Where there are alternative methods for designing the program and the programmer chooses one design arbitrarily, there is protectable expression.\textsuperscript{96} Where, in all practicality, it is necessary to design the program in a particular way, expression is merged with idea and is unprotectable.\textsuperscript{97} And of course, the function or purpose of the program is the unprotectable idea.\textsuperscript{98}

2. \textit{Literal or Nonliteral Copying}

Look and feel cases and structure, sequence, and organization cases may separate idea from expression in the same way. There is,

\begin{itemize}
\item The use of a screen to reflect the status of the program is an 'idea;' the use of a command driven program is an 'idea;' and the typing of two symbols to activate a specific command is an 'idea.' All of these elements relate to how the computer program receives commands or instructions from the user and how operationally the computer program reflects the results of those commands. \textit{Id.} at 459.
\item 92. \textit{Id.}
\item 93. "The arrangement of the parameter/command terms has no relation to how the computer operates . . . . Likewise, the highlighting and capitalizing of two specific letters of the parameter/command terms listed on the status screen has no relation to how the status screen functions . . . ." \textit{Id.} Therefore, these aspects of the computer program were not necessary incidents to the idea behind the program.
\item 94. The copyrighted program could not have been expressed in a large variety of ways . . . . The modes of expression chosen by the plaintiff for its status screen are clearly not necessary to the idea of the status screen. Therefore, the plaintiff's mode of expression of the status screen does not merge with the idea of the status screen. \textit{Id.} at 460.
\item 95. 797 F.2d 1222.
\item 96. \textit{Softklone}, 659 F. Supp. at 460 ("arrangement of the status screen involves considerable stylistic creativity . . . the defendants could have used a wide variety of techniques"); \textit{Broderbund}, 648 F. Supp. at 1132 ("there do exist other, quite different ways of expressing the ideas").
\item 97. \textit{Softklone}, 659 F. Supp. at 460 ("the modes of expression chosen by the plaintiff for its status screen are clearly not necessary to the idea . . . plaintiff's mode of expression of the status screen does not merge with the idea"); \textit{Broderbund}, 648 F. Supp. at 1132 (expression does not merge with idea because of alternative ways of expressing the idea).
\item 98. \textit{Softklone}, 659 F. Supp. at 458 ("'idea' is the process or manner by which the status screen . . . operates"); \textit{Broderbund}, 648 F. Supp. at 1132 (functions underlying the program are the idea).
\end{itemize}
however, one crucial difference between the two types of cases. An infringer may borrow the look and feel without ever copying the program. Video screens can be duplicated by simply observing the operation of a program. If the look and feel is taken in this way, the infringer has not borrowed a nonliteral aspect of the program. Rather, the infringer has literally copied the video screens.

*Broderbund* did not address this issue, but perhaps the court did not need to address the issue. In *Broderbund*, there was no question that Unison used the copyrighted program in duplicating the look and feel of the program. The fact that Unison used either the literal or nonliteral substance of the copyrighted program should be sufficient to predicate a finding of infringement of the program.

The *Softklone* court did address this issue and held that “copyright protection [of computer software] does not extend to the program screen displays. . . and does not state a claim of infringement.” Instead, the court predicated infringement on the copyrighted “status screen.” The court refused to find infringement of the program because “the same screen can be created by a variety of separate and independent computer programs,” and therefore, “[i]t is somewhat illogical to conclude that a screen can be a ‘copy’ of many different programs.” The flaw in the court’s logic, however, is that the screen may in fact have been copied from one particular program as was the case in *Broderbund*. In *Softklone*, there was no evidence that the defendant used the copyrighted program. When the infringer actually uses the program, borrowing the look and feel is nonliteral infringement of the program. However, when the infringer uses only the program’s visual displays, borrowing the look and feel is literal infringement of the video screens.

Regardless of whether look and feel is the nonliteral aspect of the program or the literal aspect of the screens, the idea/expression dichotomy is at issue since screens are a type of “form.” Blank forms are only copyrightable to the extent that they contain creative

---

99. This was precisely the situation in *Softklone*. *Id.* at 453.

100. In *Softklone*, the court did not find that the defendant infringed the copyrighted program but did find that the defendant infringed the copyrighted status screen. *Id.* at 465.

101. The defendants had access to the source code pursuant to a licensing agreement between the plaintiff and defendant. *Broderbund*, 648 F. Supp. at 1130-31.


103. *Id.* at 465.

104. *Id.* at 455-56.


106. *Softklone*, 659 F. Supp. at 460-62. Examples of blank forms include time cards, graph paper, account books, diaries, bank checks, score cards, address books, report forms, order forms and the like, designed for recording information. *Id.* at 461.
expression. The court must separate the idea underlying the screen from the particular expression used in the screen.

D. Combining the Test for Proving Copyright Infringement With the Standard for Separating Idea from Expression

The most recent cases involving copyright infringement of computer software have presented difficult idea/expression issues. The preceding analysis of these cases reveals that a programmer is free to use those aspects of another's program that are necessary to the creation of the new program but is forbidden to use those aspects of another's program that were arbitrarily chosen from the various means of designing such a program.

The Arnstein two-step approach for proving copyright infringement may once again be restated in light of this standard for separating idea from expression. First, the expert witness testifies as to the similarities in the two works and the technical reasons for the similarities. Second, the ordinary observer determines whether the similarities were caused by necessary or unnecessary programming decisions and whether the unnecessary similarities are substantial in relation to the work as a whole. As an evidentiary matter, this analysis of proving copyright infringement of computer software does not assign the ordinary observer any insurmountable tasks despite the complexities of the idea/expression dichotomy.

107. Creative expression refers to those works which provide the user with information beyond the simple indication of where to find data. Id.
108. See supra note 77 and accompanying text.
109. See supra notes 34-45 and accompanying text.
110. The method for separating idea from expression as described in this Note has been referred to as a "plurality test." Expression is protectable if it can be expressed in a plurality of ways. Note, "Look and Feel", supra note 2, at 427-28.

The plurality of expressions test combines several distinct advantages. First, the test is clear and relatively easy to apply. Instead of determining a cutoff between levels of abstraction, this test simply asks whether an idea can be expressed in a plurality of manners. Thus, the test is based on tangible and discreet factors instead of vague abstractions.

Second, this test is likely to yield more consistent and predictable results. By focusing attention on the various ways to express an idea, one generally reaches a few possible conclusions. If only one way to express an idea exists then the law provides no copyright protection. When the list of possible expressions is infinite, copyright law should permit protection, but when there are limited possibilities, the law should not provide copyright protection.

Most importantly, the plurality test incorporates the idea/expression analysis and other underlying policies of copyright law. A copyright is a limited monopoly over a particular expression. Therefore, even if the author monopolizes one expression, the plurality test allows healthy competition because viable, alternative ways to express the same idea still exist.

Id. at 428-29 (footnotes omitted).
IV. THE STANDARD FOR SEPARATING IDEA FROM EXPRESSION IS CONSISTENT WITH POLICY CONSIDERATIONS

In evaluating the policy considerations of the idea/expression dichotomy, it should initially be noted that, "in the copyright lexicon, 'idea' is no more than a metaphor for elements generally belonging in the public domain." 111 Overprotection is the result of "judicially created monopolies" 112 in works which should be dedicated to the public domain. Therefore, if a standard caused courts to protect the idea used in a computer program or the expression merged with the idea, the standard would lead to overprotection. 113 The standard advocated in this Note does not protect aspects of the program that

111. Note, Infringement of Copyright in Computer Programs, 47 U. Pitt. L. REV. 1119, 1126 (1986).

112. Judicially created monopolies exist when courts extend copyright protection to works which should be unprotected under section 102(b). As a result, the copyright owner acquires exclusive rights in the work under section 106.

113. Some commentators believe that any standard which protects nonliteral aspects of a program inherently leads to overprotection. It has been argued that protection of nonliteral aspects of computer programs stifles innovation. The thinking is that "[o]ne cannot simply 'approximate' the entire copyrighted computer program and create a similar operative program without the expenditure of almost the same amount of time as the original programmer expended." Note, Substantial Similarity, supra note 39, at 1290. Therefore, the imitator's work is innovative and it should not be discouraged. The assumption is not true. For example, in SAS the court found that where an imitator copied another's structure, sequence, and organization, the imitator's program was developed in a fraction of the time required to develop the original. SAS, 605 F. Supp. at 826. In any event, the fact that it will take a great deal of effort to copy a copyrighted work does not mean that the copier is not an infringer.

It is also thought that computer programs should not be protected because they are functional and beyond the scope of copyright. The premise is that "[c]omputer programs are fundamentally different from most other literary works that produce technological growth in that the program itself both expresses the innovation and performs the new operation." Note, Substantial Similarity, supra note 39, at 1292. According to this argument, a programmer is not able to apply the teachings of the copyrighted program in contrast to other literary works, like books, where one can apply its teachings without fear of infringement. Hence, innovation and development are stifled. While there may be a factual distinction between computer programs and other literary works, computer software infringement cases do not involve innovative imitators.

Typically the imitator will merely convert the program from one computer system to another. See, e.g., Q-Co, 625 F. Supp. 608 (S.D.N.Y. 1985) (conversion of text prompting program, from Atari to IBM computer system); E.F. Johnson Co. v. Uniden Corp. of America, 623 F. Supp. 1485 (D. Minn. 1985) (conversion of LTR-compatible radio program); Whelan, 609 F. Supp. 1307 (conversion of dental-business program from large IBM to small IBM system); SAS, 605 F. Supp. 816 (conversion of statistical analysis program, from IBM to VAX computer system). Furthermore, "the copyright law has always recognized and tried to accommodate the fact that all intellectual pioneers build on the work of their predecessors. Thus, copyright principles derived from other areas are applicable in the field of computer programs." Whelan, 797 F.2d at 1238 (criticizing Note, Substantial Similarity, supra note 39). Lastly, the standard advocated in this Note does not protect aspects of a com-
belong in the public domain. It provides protection for the programmer's creative efforts while promoting dissemination of programming techniques needed to implement new programming ideas. Necessary programming techniques must remain in the public domain if the computer industry is to continue to be innovative.

A. New Programming Techniques Remain in the Public Domain

In order to evaluate the works that belong in the public domain, it is useful to compare copyright protection to patent law. Patent and copyright laws derive their power from the Constitution and are enacted by federal statutes. Together, these forms of protection define the public domain.

Some aspects of a computer program may be dedicated to the public domain. A computer program is primarily a process, a set of instructions which bring about a certain result. Processes are excluded from copyright protection and, additionally, may be dedicated to the public domain as uninventive patentable subject matter. A work is dedicated to the public domain by operation of patent law when it is patentable subject matter but does not meet the high standards of inventiveness required to be patented. A standard for separating idea from expression which provides copyright protection for the process aspect of a computer program would conflict with patent law by protecting works which may be dedicated to the public domain.

114. U.S. Const. art. I, § 8, cl. 8. Article I, § 8, cl. 8 gives Congress the power "[t]o promote the Progress of Science and useful Arts, by securing for limited Times to Authors and Inventors the exclusive Right to their respective Writings and Discoveries."


117. 17 U.S.C. § 102(b) (1982). "Section 102(b) is intended, among other things, to make clear that the expression adopted by the programmer is the copyrightable element in a computer program, and that the actual processes or methods embodied in the program are not within the scope of the copyright law." CONTU, Final Report, supra note 11, at 19 (quoting U.S., Congress, Senate, Judiciary Committee, 94th Cong., 1st Sess., 1975, S. Rep. No. 54) (emphasis in original).

Furthermore, copyright protection of a process would convert the Copyright Act into a general misappropriation statute. Copyright is a type of misappropriation statute protecting against the improper appropriation of a select set of intellectual works. The framers' intent to prevent the creation of a general misappropriation statute is clear from section 102(b) of the Copyright Act and the policy of promoting the widest possible dissemination of ideas.

The standard advocated in this Note does not protect the process aspect of a computer program. Under the standard, a programmer is allowed to borrow any aspect of another's program which is needed to automate the process in the copyrighted program or a similar process. The programmer is prohibited from using only the technically unneeded creative efforts of another.

In addition, the standard promotes a desirable economic policy. "One of the hallmarks of a competitive industry is the ease with which entrepreneurs may enter into competition with firms already doing business. The absence of significant barriers to entering the program-writing market is striking." The competitive process would be injured if programmers could hoard their technological advantages. This is not the case. Any necessary programming technique which is not patented or held as a trade secret is freely accessible. The standard prevents only wholesale-type copying of another's creative work product. This type of copying is counter-competitive since it allows an infringer to drastically reduce production costs and reap unearned profits.

B. Two Instances of Overprotection

To the extent that courts have separated idea from expression

119. Nimmer warned of this result in CONTU, FINAL REPORT, supra note 9:
If literary works are to be so broadly construed, the Copyright Act becomes a general misappropriation law, applicable as well in what has traditionally been regarded as the patent arena, and, indeed, also in other areas to which neither copyright nor patent law has previously extended. This poses a serious constitutional issue in that it is arguable that such an approach stretches the meaning of authors and writings as used in the Copyright Clause of the Constitution beyond the breaking point.

Id. at 26 (emphasis in original).

120. The set of selected works is defined in 17 U.S.C. § 102(a). See supra note 14 and accompanying text.

121. Section 102(b) of the Copyright Act explicitly limits copyright protection against general misappropriation of things like ideas, functions and processes.

122. See supra note 2 and accompanying text.

123. See Whalen, 797 F.2d at 1236. See also supra notes 95-98 and accompanying text.

124. CONTU, FINAL REPORT, supra note 11, at 23.

125. SAS, 605 F. Supp. at 826 ("by expropriating a completed and proven design, [the defendant] was saved the substantial time and effort of developing its own original design").
consistent with the standard advocated in this Note, current trends in software protection have not lead to overprotection. In the two instances that follow, the courts did not comply with the standard and, as a result, protected works that belong in the public domain. It should be noted that, while these are not pure computer software cases, they do involve related subject matter.

In *Williams v. Arndt*, the court extended copyright protection to a method of operation. Williams developed a method for trading in various commodities and published a booklet entitled *Floor Trader's Method*. Arndt later used this method in a computer program. The court found that the program and the booklet used many of the same procedures and produced the same results. The court further found that designing the program was merely a clerical task lacking in creative expression. Based on these findings, the court held that Arndt's computer program infringed upon the copyrighted booklet.

Arndt should not have been prohibited from borrowing the method of trading. The Copyright Act expressly excludes methods of operation and processes from copyrightable subject matter. Therefore, the court's finding, that the same procedures were used and the same results produced, does not imply that Arndt borrowed protectable expression. These aspects of the copyrighted work were closely linked to the idea underlying the *Floor Trader's Method* and would, therefore, be unprotectable under the standard for separating ideas from expression.

The court's second finding, that the programming was merely a clerical function lacking in expression, is irrelevant. The inquiry is not whether the allegedly infringing work contains protectable ex-

---

127. *Id.* at 579.
128. *Id.* at 573.
129. *Id.* at 574.
130. *Id.* at 576.
131. *Id.* at 579.
132. *Id.* As stated by the court:
Writing the program may be a task requiring a great deal of inventiveness, or it may be merely a clerical function. To a skilled programmer, the conversion of known input, known output, the mathematical expressions needed and the methods of transferring those expressions into computer language is necessarily a mere clerical function.

*Id.* at 577.
133. *Id.* at 581.
135. *See supra* notes 95-98 and accompanying text.
136. The court concluded that "the source code in this case is not a different and unique expression of a well known idea." *Williams*, 626 F. Supp. at 578. The uniqueness of the defendant's work is irrelevant. In fact, the court's statement tends to imply that the defendant borrowed only ideas from the copyrighted work.
pression. Rather, the issue is whether the alleged infringer borrowed a substantial amount of another's protectable expression.137

Similarly, copyright protection was extended to the functional aspect of an animated toy in the recent group of cases collectively known as the Teddy Ruxpin Cases.138 Teddy Ruxpin is a toy bear whose eyes, nose, and mouth move in synchronization with his voice when a specially designed cassette tape is inserted in his back.139 Worlds of Wonder (WOW) copyrighted Teddy Ruxpin as an audiovisual work.140 Two separate infringement suits were brought against manufacturers of tapes that activated Teddy Ruxpin. In both cases, the courts granted the temporary injunctions finding likelihood of infringement of the copyright on Teddy Ruxpin because the tapes duplicated the look and feel of the audiovisual effect of Teddy Ruxpin.141

Surprisingly, one court did not accept the argument made by the defendant that the copyright should have been on the tapes—not on Teddy Ruxpin.142 Teddy Ruxpin is merely a sophisticated tape player. It is far beyond the scope of copyright to protect a mechanical device which generates sound and image.143

Even assuming, arguendo, that infringement may be predicated on the Teddy Ruxpin copyright, the defendants should not have been prohibited from activating the motors in the face of the toy. Applying the standard for separating idea and expression, the idea is an animated bear. Causing the bear to be animated by activating Teddy Ruxpin's eyes, nose, and mouth is a necessary incident to the idea and therefore is unprotectable.144 The defendants should only have been prohibited from using the same or substantially similar sequence of movements. Also, it would be appropriate, as one of the courts noted, to prohibit the use of the Teddy Ruxpin voice and story theme.145 Under the standard, these are arbitrary choices of expression because many other voices and stories are available.146

137. See supra note 31 and accompanying text.
142. See Veritel, 658 F. Supp. at 354-55. This issue was not raised in the Vector case.
143. If it is valid to copyright Teddy Ruxpin as an audiovisual work, why not copyright a VCR and prevent others from producing VHS tapes? The only difference is that a VCR can create a wider variety of visual images when connected to a TV.
144. See supra notes 95-98 and accompanying text.
145. Vector, 653 F. Supp. at 140.
146. See supra notes 95-98 and accompanying text.
In both Arndt and the Teddy Ruxpin cases, the works were appropriate subject matter for a utility patent on a process. The failure to obtain patent protection dedicated the works to the public domain. Logical extremes of Arndt and the Teddy Ruxpin cases lead to bizarre results because copyright protection was extended to works in the public domain. For example, some computers are loaded with a cassette tape containing encoded sounds. If the tape caused the computer to create images on the screen which told stories to children, would the Courts in the Teddy Ruxpin cases allow the manufacturers to obtain a copyright on the computer and thereby prevent others from producing cassettes which activated the computer? What if the computer were designed as a soft sculpture toy? It is unclear what precedential value these cases will have on computer software litigation.

CONCLUSION

This Note discussed the creative programming efforts associated with designing nonliteral aspects of a computer program—the structure, sequence, and organization and the look and feel. Given the policy of the Copyright Act to encourage production of creative and innovative works, there is no question that nonliteral aspects of the computer program should be protected. The question is how to prove copyright infringement of the computer program when the alleged infringer borrowed the nonliteral aspect. The troublesome issue is whether the alleged infringer borrowed idea or expression. An ordinary observer without any special understanding of computer programming must make the determination.

This Note advocates a simple standard for separating idea from expression which the ordinary observer can apply in the Arnstein two-step infringement test. In the first step, the court admits expert testimony to evidence similarities in the two works and the technical reasons for the similarities. In the second step, the ordinary observer considers the expert testimony and separates similarities in idea from similarities in expression. The ordinary observer then determines whether the similarities in expression are a substantial part of the copyrighted work.

The standard advocated in this Note aids the ordinary observer in the separation of idea from expression. Where there are alternative

148. See supra note 118 and accompanying text.
149. See supra notes 51-61 and accompanying text.
150. See supra note 2 and accompanying text.
151. See supra notes 32-33 and accompanying text.
methods for designing the program and the programmer chooses one design arbitrarily, there is protectable expression. Where, in all practicality, it is necessary to design the program in a particular way, expression is merged with idea and is unprotectable. The function or purpose of the program is the unprotectable idea.

Application of this standard is desirable for two reasons. First, it facilitates a workable method for proving copyright infringement. And second, it is consistent with the policies of the Copyright Act: providing protection for the programmer’s creative efforts while promoting dissemination of programming techniques needed to implement new programming ideas.

George N. Grammas