Washington and Lee Law Review

Volume 48 | Issue 3 Article 8

Summer 6-1-1991

Determining The Scope Of Copyright Protection For A Computer Program'S Nonliteral Elements: Is It As Easy As 1*2*3*?*

NOTES

DETERMINING THE SCOPE OF COPYRIGHT PROTECTION FOR A COMPUTER PROGRAM'S NONLITERAL ELEMENTS: IS IT AS EASY AS 1*2*3*?*

The Copyright Act of 1976[1] provides copyright protection to computer programs.[2] However, the Copyright Act does not provide unlimited protection to computer programs.[3] Much confusion and debate among courts,[4]

---

* A version of this Article was submitted in the Nathan Burkan Memorial Competition.

1. Copyright Act of 1976, 17 U.S.C. §§ 101-810 (1990) [hereinafter Copyright Act].

2. *See* Copyright Act, *supra* note 1, § 101 (defining computer program as used in Copyright Act). The Copyright Act defines a computer program as a set of statements or instructions that a computer uses directly or indirectly to bring about a certain result. *Id.*; *see also* L. KUTTEN, COMPUTER SOFTWARE PROTECTION/LIABILITY/LAW/FORMS §§ 2.01[1]-[11] (1990) (discussing copyright law and specific inclusion of computer programs in Copyright Act).

3. *See* H.R. REP. No. 1476, 94th Cong., 2d Sess. 46, 54 (1976), *reprinted in* 1976 U.S. CODE CONG. & ADMIN. NEWS 5659, 5667 [hereinafter HOUSE REPORT] (stating that computer programs fall within Copyright Act's definition of literary works). Under the Copyright Act, Congress classified computer programs as a form of literary work. *Id.* The Copyright Act defines literary works as "works, other than audiovisual works, expressed in words, numbers, or other verbal or numerical symbols or indicia, regardless of the nature of the material objects . . . in which they are embodied." Copyright Act, *supra* note 1, § 101. However, the Copyright Act limits the copyright protection available to any copyrighted subject matter. *See id.* § 102(b) (stating that copyright protection does not extend beyond certain boundaries). According to section 102(b) of the Copyright Act, copyright protection for an original work of authorship does not extend to "any idea, procedure, process, system, method of operation, concept, principle, or discovery, regardless of the form in which it is described, explained, illustrated, or embodied in such work." *Id.*

Courts also have held that computer game video displays are copyrightable as audiovisual works. *See, e.g.*, M. Kramer Mfg. Co. v. Andrews, 783 F.2d 421, 435-36 (4th Cir. 1986) (holding that copyright protects audiovisual screens of computer video games); Atari v. North Am. Phillips Consumer Elecs. Corp., 672 F.2d 607, 617 (7th Cir.), *cert. denied*, 459 U.S. 880 (1982) (same); Atari v. Amusement World, 547 F. Supp. 222, 226 (D. Md. 1981) (same).

4. *See generally* Synercom Technology v. University Computing, 462 F. Supp. 1003, 1012-13 (N.D. Tex. 1978) (holding that copyright does not protect particular sequence of data in computer program); Apple Computer v. Franklin Computer, 714 F.2d 1240, 1249 (3d Cir. 1983) (holding that copyright protects computer program object code and source code), *cert. dismissed*, 464 U.S. 1033 (1984); Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1239 (3d Cir. 1986) (holding that computer program copyright protection extends beyond object code and source code to structure, sequence, and organization of code), *cert denied*, 479 U.S. 1031 (1987); Plains Cotton Co-op Ass'n v. Goodpasture Computer Service, 807 F.2d 1256, 1262 (5th Cir.) (declining to embrace *Whelan* and holding that copyright does not protect structure, sequence, and organization of computer program code), *cert. denied*, 484 U.S. 821 (1987); Broderbund Software v. Unison World, 648 F. Supp. 1127, 1133 (N.D. Cal. 1986) (embracing *Whelan* and extending copyright protection to structure, sequence, and organization of computer program audiovisual menu screens); Digital Communications v. Softklone Distrib.

scholars,[5] and industry participants[6] has resulted from attempts to define

---

Corp., 659 F. Supp. 449, 456 (N.D. Ga. 1987) (rejecting *Broderbund* extension of copyright protection to computer screens and holding that copyright protects computer program screens only when programmer obtains separate copyright registration on computer screens); Manufacturers Technologies v. Cams, 706 F. Supp. 984, 992-93 (D. Conn. 1989) (stating that copyright registration changes alter viability of *Digital* decision and holding that copyright protection of computer program extends to computer program screens under legal fiction of two copyright registrations, one for program code and one for program screens, existing in single copyright registration on computer program); Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 68 (D. Mass. 1990) (stating that copyright protection of computer program extends to structure, sequence, and organization of computer program's menu command structure).

5. *See generally* Nimmer and Krauthaus, *Copyright and Software Technology Infringement: Defining Third Party Development Rights*, 62 IND. L.J. 13 (1986) (stating that copyright protection of computer programs must balance needs of original author and second, value-added user and that certainty in balancing these needs in software infringement suits may be impossible); Note, *"Look and Feel" as A Copyrightable Element: The Legacy of* Whelan v. Jaslow?, 51 LA. L. REV. 177 (1990) (stating that once plaintiff establishes prima facie case in copyright infringement suit, defendant should bear burden of proving that similarities between programs are not due to any proscribed activity rather than plaintiff bearing burden of proving copyrightability of similar computer program elements through strained abstractions analysis); Note, *Idea, Process, or Protected Expression?: Determining the Scope of Copyright Protection of the Structure of Computer Programs*, 88 MICH. L. REV. 866 (1990) (stating that analysis of copyright protection for computer program structure must include process or expression distinction in addition to idea or expression distinction and merger doctrine); Note, *Copyright Protection for Computer Screen Displays*, 72 MINN. L. REV. 1123 (1988) (stating that courts should extend copyright protection to computer program screens but use heightened originality standard, concrete, work-specific definition of screen's idea, and less grudging application of merger doctrine); Note, *Defining the Scope of Copyright Protection for Computer Software*, 38 STAN. L. REV. 497 (1986) (stating that courts judging similarities between two computer screen displays should consider them as conventional audiovisual works, and courts judging similarities in two computer program codes should consider structure of programs in addition to individual lines of code); Note, *Screen Displays are Proper Subject Matter for Copyright Protection*, 1988 U. ILL. L. REV. 757 (1988) (stating that Congress should review copyright protection of computer screen displays and set explicit legislative guidelines to define scope of copyright protection for computer screen displays); Note, Whelan Associates v. Jaslow Dental Laboratories: *Copyright Protection for Computer Software Structure - What's the Purpose?*, 1987 WIS. L. REV. 859 (1987) (stating that copyright protection of computer programs will result in duplication of logic development costs, adequate economic incentives already exist to promote development of software, and courts should strictly limit holding in *Whelan*); Note, *The Scope of Copyright Protection for Computer Programs: Exploring the Idea/Expression Dichotomy*, 43 WASH. & LEE L. REV. 1373 (1986) (stating that federal judiciary should continue to expand scope of copyright protection for computer programs to encompass programs' user interface, structure, and organization, thereby promoting Congress' desire to foster creative and intellectual achievements in computer programming through copyright laws); Comment, *A Rose By Any Other Name: Computer Programs and the Idea-Expression Distinction*, 34 EMORY L.J. 741 (1985) (discussing importance of idea or expression distinction and suggesting shortening life of copyright protection for computer programs while expanding scope of copyright protection for computer programs); Comment, *Softright: A Legislative Solution to the Problem of Users' and Producers' Rights in Computer Software*, 44 LA. L. REV. 1413 (1984) (stating that software protection legislation can eliminate problems of software protection and copying while furthering interests of developers and users); Comment, *Copyright Protection for Computer Software after* Whelan Associates v. Jaslow Dental Laboratory, 54 MO. L. REV. 121 (1989) (stating that settled issues of computer copyright protection include object code

the scope of the Act's limited copyright protection for computer programs.[7] Nevertheless, by adopting a common framework to separate the protected parts of a computer program from the unprotected parts, the courts can eliminate much of this confusion and debate.[8]

To appreciate the difficulty in determining the scope of copyright protection for any given program, an understanding of how programmers develop computer programs and how computers operate is necessary. A programmer creates a computer program in stages that successively become more detailed and specific.[9] First, the programmer must identify and define the objective that the program is to accomplish.[10] As the definition process progresses, the programmer collects information about the objective and develops a general understanding of how to accomplish the objective.[11] The

---

and source code, but unsettled issues of computer copyright protection include scope of copyright protection and proper test for infringement claims); Comment, *The Incompatibility of Copyright and Computer Software: An Economic Evaluation and a Proposal for a Marketplace Solution*, 66 N.C.L. REV. 977 (1988) (stating that ad hoc copyright protection for computer programs has produced more protection than is necessary to provide incentive to create and recommending marketplace approach to software protection utilizing contract law as foundation); Comment, Manufacturers Technologies, Inc. v. Cams, Inc. - *The Legal Fiction Created by a Single Copyright Registration of a Computer Program and Its Display Screens*, 65 NOTRE DAME L. REV. 536 (1990) (stating that legal fiction approach to copyright infringement suits is not proper framework for computer copyright analysis).

6. *See Registration of Computer Screens is Subject of Copyright Office Hearing*, Pat. Trademark & Copyright J. (BNA) No. 847, at 507 (Sept. 17, 1987) (discussing responses by various industry participants to proposed changes in registration requirements for computer programs). Apple Computer urged the Copyright Office to maintain separate copyright registrations for a computer program and a computer program's audiovisual displays. *Id.* This separation will clarify that the scope of copyright protection applicable to the literary work embodied in the program code differs from the scope of copyright protection applicable to the audiovisual work embodied in the screen displays. *Id.* Lotus Development urged the Copyright Office to continue to offer only one copyright registration for each computer program. *Id.* at 508. This single registration can encompass all of the copyrightable elements contained in the program, and the single registration will provide the flexibility necessary for software development. *Id.*

7. *See infra* notes 56-163 and accompanying text (describing court conclusions in attempts to identify copyrightable elements of computer programs).

8. *See infra* notes 168-224 and accompanying text (proposing *Lotus* framework as common framework for defining computer program protection under Copyright Act).

9. *See* Menell, *An Analysis of the Scope of Copyright Protection for Application Programs*, 41 STAN. L. REV. 1045, 1051 (1989) (stating that often programmers develop programs in five stages: defining desired task; flowcharting; coding; debugging; preparing documentation); Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1229 (3d Cir. 1986) (stating that creating program often takes several steps, moving from general steps to specific steps), *cert. denied*, 479 U.S. 1031 (1987).

10. *See* Menell, *supra* note 9, at 1052 (stating that programmer first must identify user tasks that program will accomplish). In defining the task that the program will accomplish, a programmer must consider the needs and abilities of the end users and the particular limitations and attributes of their work environment. *Id.* Therefore, a programmer must spend a considerable amount of time in the design phase understanding the users of the program and determining the best way to serve their needs. *Id.*; *see also Whelan*, 797 F.2d at 1229 (stating that first step for programmer is identifying problem that program will solve).

11. *See* Menell, *supra* note 9, at 1052 (stating that programmer must understand users

programmer then formulates this general understanding into a flowchart that is composed of a logical series of steps describing how to accomplish the objective.[12] After completing the initial flowchart, the programmer continues to subdivide each section of the flowchart into a series of smaller tasks that become increasingly simple.[13] In this manner a programmer transforms a complex objective into a series of small tasks that operate together to accomplish the overall objective.[14]

The process of flowcharting and reducing the objective to a series of smaller tasks is not an exact science.[15] The identification and arrangement of these smaller tasks control the overall operation of the program, including how the program communicates with the end user, receives data from the end user, and manipulates the data to achieve a result.[16] How effectively the program accomplishes each of these tasks depends on the expectations of the end user and the knowledge and talent of the programmer.[17]

---

and their needs). During a program's design phase, a programmer usually researches the nature of the task that the program will accomplish as well as the backgrounds and abilities of the users and the attributes of the users' work environment. *Id.* The computer industry refers to this process as the study of human factors and defines this study as the field of computer-human interaction. *Id.* at 1053. Although relatively new, the field of computer-human interaction has impacted greatly on the development of application programs. *Id.* The main impact has been in the way that a programmer both conceptualizes and writes an application program. *Id.* As a programmer conceptualizes and writes a program, a programmer strives to incorporate the following five human factor goals into the program's design: minimize the user's learning time; maximize the user's speed of performance; minimize the user's rate of errors; maximize the user's satisfaction; maximize the user's retention of knowledge over time. *Id.* at 1053-54; *see generally* Curtis, *Engineering Computer "Look and Feel": User Interface Technology and Human Factors Engineering*, 30 JURIMETRICS J. 51 (1989) (discussing history of user interface design and impact of human factors engineering on user interface design).

12. *See Whelan*, 797 F.2d at 1230 (stating that programmer usually outlines solution in flowchart form as programmer learns more about problem).

13. *See* Menell, *supra* note 9, at 1055 (stating that most common form of flowcharting is top down approach). The top down approach facilitates the conversion of large, complex problems into smaller, less complex problems. *Id.* Utilizing the top down approach, the programmer breaks down each task into a series of modules or subroutines, and then, the programmer breaks the subroutines down into individual component parts. *Id.* This process continues until the programmer can translate the components of the flowchart into a programming language. *Id.*

14. *Id.*

15. *See Whelan*, 797 F.2d at 1230 (stating that programmer has numerous ways to solve data organization problems that may arise). Each solution to define data organization problems may have particular characteristics that make it more or less desirable from the perspective of the overall program. *Id.*

16. *See id.* (stating that as programmer refines program structure, programmer must decide what data program needs, when user inputs data, how user inputs data, and how program combines data with other data).

17. *See* Menell, *supra* note 9, at 1056 (stating that although some programmers have distinctive programming styles, common goal is to produce program that is accurate, efficient, and reliable). Several factors comprise the concept of computer program efficiency: code efficiency, which is maximizing the program's processing speed; memory efficiency, which is minimizing the amount of computer memory necessary to perform the desired tasks; input/

After defining the program structure, the programmer, [by coding the program] converts the program structure into a form that the computer can understand.[18] This conversion has two possible forms.[19] The first form, known as object code, is the only code that a computer can execute directly.[20] Because writing programs in object code is very cumbersome, most programmers write programs in a second code form known as source code.[21] A source code is a computer "language" that has specific commands and syntax, but a source code permits programmers to write commands in a simpler format.[22] The computer has the capability, through the use of a compiler or an interpreter, to convert the source code into an object code and then execute the instructions.[23] After completing the coding, the programmer runs the program to detect and to correct any errors.[24] After correcting the errors, the programmer's only remaining duties are documenting how to operate the program and performing any maintenance that the program may require.[25]

The actual coding is a relatively small part of the overall computer program development process.[26] Typically, only twenty percent of a program's development costs are for coding.[27] The remaining eighty percent of

---

output efficiency, which is maximizing the quality and speed of information transmission between the computer and the user. *Id.* at 1052-54.

18. *See* Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1230 (3d Cir. 1986) (stating that after programmer completes design of program, programmer converts subroutines into language computer can understand), *cert. denied*, 479 U.S. 1031 (1987).

19. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 43 (D. Mass. 1990) (describing object code and source code as two coding forms that represent literal or written manifestation of computer program).

20. *See id.* at 44 (stating that object code is language which computer can execute directly without translation).

21. *See id.* (noting that object code rarely is written directly by computer programmers).

22. *See id.* (stating that source code is computer program written in some programming language, such as FORTRAN (*FOR*mula *TRAN*slation), COBOL (*CO*mmon *B*usiness *O*riented *L*anguage), Pascal, or BASIC, that uses complex symbolic names, along with complex rules of syntax). The *Lotus* court compared object code and source code through the following task: "divide the value of 'B' by the value of 'C' and add the result to the value of 'A.'" *Id.* A programmer writing in object code might accomplish this task as follows: 0010000000010001; 1000000011010010; 1101000000010000. *Id.* However, a programmer writing in source code might accomplish the same task as A + B/C. *Id.*

23. *See id.* (stating that either compiler translates entire program before computer executes program or interpreter translates each program line simultaneously as program executes line).

24. *See* Menell, *supra* note 9, at 1051 (stating that fourth stage of computer program development involves testing program for accuracy, correcting programming errors, and verifying that program functions properly).

25. *See* Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1231 (3d Cir. 1986) (stating that documentation and maintenance are important in creating computer programs), *cert. denied*, 479 U.S. 1031 (1987).

26. *See id.* (stating that larger portion of expense and difficulty in creating computer programs is attributable to development of structure and logic of program, rather than to coding). The development of the structure and logic of a computer program includes debugging, documentation, and maintenance. *Id.*

27. *See id.* (noting that only 20% of program development cost goes into encoding).

the development costs are for the design of the program.[28] In addition to representing most of the development costs, the design of the program also determines the commercial success or failure of the program because the end user judges a computer program on the strength of its design and operation.[29] Therefore, the design is the most important and valuable part of a program to a computer program developer.[30]

Consequently, the design is the portion of the program that a programmer wants to protect through copyright. However, doubt remains as to whether the scope of copyright protection extends this far. The scope of copyright protection for a given copyrighted work is only at issue when the owner of the copyright registration claims an infringement.[31] In an infringement suit, the copyright holder must establish ownership of a valid copyright on the allegedly infringed work.[32] After establishing ownership of a valid copyright, the copyright holder must prove that the allegedly infringing work is a copy of the copyrighted work.[33] Ownership of a copyright registration is prima facie evidence of a valid copyright,[34] but ownership merely raises a rebuttable presumption.[35] The owner of the allegedly infringing work can rebut this presumption by proving that copyright law does not protect the portions of the copyrighted work allegedly infringed.[36]

---

28. *See supra* notes 9-17 and accompanying text (describing most cost intensive steps in developing computer program).

29. *See* Curtis, *supra* note 11, at 52-53 (describing importance of computer program's user interface design to commercial success of program). The design of the computer program's user interface is crucial in marketing the program to end users. *Id.* End users quickly judge a program's ease of use and learning through the program's user interface. *Id.* A superior user interface can differentiate a computer program from competing computer programs. *Id.*

30. *See id.* (stating that user interface has become paramount concern in designing computer programs).

31. *See infra* notes 32-36 and accompanying text (describing elements of copyright infringement suit); L. KUTTEN, *supra* note 2, §§ 2.07[1]-[5] (discussing copyright elements, defenses, and remedies of copyright infringement suits).

32. *See* Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1231 (3d Cir. 1986) (holding that to prove claim of copyright infringement, copyright holder bears burden of proving ownership of valid copyright), *cert. denied*, 479 U.S. 1031 (1987).

33. *See id.* (holding that copyright owner must prove that allegedly infringing work is copy of copyrighted work).

34. *See* Copyright Act, *supra* note 1, § 410 (describing issuance and effect of valid copyright registration). The Register of Copyrights will register a claim and issue a certificate of registration under the seal of the Copyright Office after examining the material presented for copyright registration, determining that the material constitutes copyrightable subject matter, and ensuring that the applicant has fulfilled all remaining legal and formal requirements of the Copyright Act. *Id.* A certificate of registration, issued prior to or within five years of the first publication of the registered work, constitutes prima facie evidence in any judicial proceeding that the copyright and the facts stated in the certificate are valid. *Id.*

35. *See infra* notes 36-41 and accompanying text (describing how defendant can rebut presumption raised by ownership of valid copyright registration).

36. *See* Digital Communications v. Softklone Distrib. Corp., 659 F. Supp. 449, 456 (N.D. Ga. 1987) (holding that where copyright owner has proven validity of copyright registration, owner of allegedly infringing computer program bears burden of producing evidence that brings into question copyrightability of original program).

The debate over what portion of a literary work copyright law protects and what portion of a literary work copyright law does not protect usually centers on the distinction between an idea and its expression.[37] Copyright law does not protect an idea but does protect the expression of an idea within certain limits.[38] First, the expression must be original.[39] Second, the expression must be separable from the idea and not essential to the idea.[40] Third, even if the expression is not essential to the idea, copyright law will not protect the expression if the expression is one of but a limited number

---

37. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 53 (D. Mass. 1990) (stating that idea or expression distinction has long been fundamental part of copyright law).

38. *See* Baker v. Selden, 101 U.S. 99, 103 (1879) (holding that copyright law does not protect ideas, but only expressions of ideas). In *Baker*, Selden owned copyright registrations on a series of books that described a peculiar system of double entry bookkeeping. *Id.* at 99-100. The books contained several blank forms, consisting of ruled lines and headings, that illustrated the system and were essential to the use of the system. *Id.* at 100. Baker subsequently published a book that described the system developed by Selden and employed similar blank forms. *Id.* In the ensuing copyright infringement suit, Selden contended that his copyright registration entitled him to the exclusive use of the forms contained in the book, and any use of similar forms in a subsequent work violated the copyright. *Id.* at 101. In addressing this contention, the Supreme Court stated that a clear distinction exists between a book and the art that the book illustrates. *Id.* at 102. A copyright registration on a book protects the author's particular explanation of the art described in the book, but the copyright registration does not automatically give the author the exclusive right to use the art described in the book. *Id.* If the forms in a book, or forms similar to them, are necessary to the performance of the art described in the book, then the forms are necessary incidents to the art. *Id.* at 103. Such necessary incidents are in the public domain, and a copyright registration does not entitle the author to the exclusive use of such necessary incidents. *Id.*

In *Mazer v. Stein*, the Supreme Court expanded the principle espoused in *Baker v. Selden*. *See* Mazer v. Stein, 347 U.S. 201, 217 (1954) (holding that copyright protects only expression of idea, not idea itself). Stein created and sold lamps with statuettes as lampbases. *Id.* at 203. Stein owned a copyright registration on the statuettes, and Mazer, without authorization, copied the statuettes and began selling lamps with the statuettes as lampbases. *Id.* at 202-03. In the ensuing copyright infringement suit, the Supreme Court held that the copyright registration did not give Stein the exclusive right to use any statuettes as lampbases. *Id.* at 217. However, the copyright registration did give Stein the exclusive right to use the registered statuettes as lampbases, and Stein could use the copyright registration to prevent others from making lamps that incorporated copies of the registered statuettes. *Id.*

Congress incorporated the distinction between idea and expression into the Copyright Act. *See* HOUSE REPORT, *supra* note 3, at 5670 (stating that § 102(b) of the Copyright Act specifically precludes copyright protection from any idea, procedure, process, system, method of operation, concept, principle, or discovery). Copyright does not grant an author exclusive use of ideas or information contained in the author's work. *Id.*

39. *See* Copyright Act, *supra* note 1, § 102(a) (stating that copyright protection subsists in original works of authorship); *see also* HOUSE REPORT, *supra* note 3, at 5664 (stating that originality and fixation in tangible form are two fundamental criteria of copyright protection). Congress purposely left the definition of the standard "original works of authorship" to the courts, but Congress specifically stated that the standard did not require novelty, ingenuity, or aesthetic merit. *Id.*; *see also Lotus*, 740 F. Supp. at 48 (stating that original refers to works independently created by author).

40. *See Lotus*, 740 F. Supp. at 52 (stating that elements of useful article that can exist independently of utilitarian aspects of article are copyrightable as elements of expression).

of possible expressions.[41] The reason for these limitations is that the primary goal of copyright protection is to benefit the public, and Congress' decision to reward authors with copyright monopolies is only a means to this end.[42] Society benefits if authors generate new ideas and disclose these ideas to the public.[43] Copyright law provides encouragement by protecting an author's work, but the protection cannot be so broad as to be detrimental to the public.[44] The Copyright Act attempts to balance these goals by refusing

---

41. *See id.* at 59 (discussing concept of merger and lack of copyright protection for expressions that are limited in number). If specific expression is one of a limited number of ways of expressing an idea, the expression is not copyrightable. *Id.* Otherwise, copyright protection easily would permit an author or authors to control an idea that has a limited number of expressions by copyrighting all of the possible expressions. *Id.*

In *Morrissey v. Proctor & Gamble* the Circuit Court of Appeals for the First Circuit espoused the basic principle underlying the concept of merger. 379 F.2d 675, 678-79 (1st Cir. 1967). The *Morrissey* court stated:

> [w]hen the uncopyrightable subject matter is very narrow, so that the 'topic necessarily requires,' if not only one form of expression, at best only a limited number, to permit copyrighting would mean that a party or parties, by copyrighting a mere handful of forms, could exhaust all possibilities of future use of the substance. In such circumstances it does not seem accurate to say that any particular form of expression comes from the subject matter. However, it is necessary to say that the subject matter would be appropriated by permitting the copyrighting of its expression. We cannot recognize copyright as a game of chess in which the public can be checkmated.

Id.

In *Herbert Rosenthal Jewelry Corp. v. Kalpakian*, the Circuit Court of Appeals for the Ninth Circuit espoused a classic application of the merger doctrine. 446 F.2d 738 (9th Cir. 1971). In *Kalpakian* the court held that copyright did not protect the idea of a jewel-encrusted bee pin. *Id.* at 742. Any manufacturer could only design a jewel-encrusted bee pin in a limited number of ways; thus, the idea of a jewel-encrusted bee pin merges with the expression of a jewel-encrusted bee pin because the expression cannot be separated from the idea. *Id.* If the court permitted copyright protection for the pin, the manufacturer that held the copyright registration would have a monopoly on the jewel-encrusted bee pin market because no other manufacturer could create a substantially different jewel-encrusted bee pin. *Id.*

42. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 52 (D. Mass. 1990) (noting that Congress grants copyright monopolies for purpose of promoting public welfare). Congress grants copyright monopolies to serve the public welfare by encouraging authors to generate new ideas and disclose these ideas to the public in any uniquely expressed way. *Id.*

43. *See id.* (stating that encouraging individual effort by personal gain is best way to advance public welfare); Mazer v. Stein, 347 U.S. 201, 219 (1954) (stating that economic philosophy behind constitutional clause empowering Congress to grant patents and copyrights is conviction that encouragement of individual effort by personal gain is best way to advance public welfare through talent of authors and inventors).

44. *See Lotus*, 740 F. Supp. at 52 (stating that competing goals of copyright protection are advancement of public welfare and reward of individual authors). Balancing the competing goals of copyright protection is a delicate task. *Id.*; *see also* Menell, *supra* note 9, at 1047 (stating that establishing a precise line between copyrightable expression of computer programs and noncopyrightable processes of computer programs is impossible). If courts construe copyright protection for computer programs too broadly then courts will grant strong monopolies to the initial authors of a program and will prohibit other authors from developing improved products. *Id.* at 1047-48. If courts construe copyright protection for computer

to draw a bright line distinction between ideas and expression and leaving this determination to the courts.[45]

As the use of computers has proliferated, the creation of computer programs has become an extremely large and lucrative business.[46] Identifying needs for computer programs and fulfilling those needs has produced great financial rewards and fostered great leaps in technological innovation.[47] The rapid innovation in computers and computer programming also has permitted many individual programmers successfully to develop and to market programs, and the programming industry has not become totally dominated by a few large manufacturers.[48] Success, however, has its price, and because of increased program copying, many problems have developed in the computer programming industry.[49] This copying in turn has generated the increasing number of lawsuits that have disputed the scope of copyright protection for computer programs.[50]

Early case law established that copyright protection of a computer program extends to its object code and source code, and consequently, copyright law now firmly protects these "literal elements."[51] Copyright

---

programs too narrowly then courts will allow other authors to copy an initial author's work easily and will discourage all but modest innovation. *Id.* at 1048.

45. *See Lotus*, 740 F. Supp. at 53 (stating that Congress has refused to draw boundary line between copyrightable and noncopyrightable elements of computer programs). Congress has mandated that courts determine the boundary line between copyrightable and noncopyrightable expression by using an evaluative standard that balances the competing goals of copyright protection. *Id.*

46. *See* A. CLAPES, SOFTWARE, COPYRIGHT, & COMPETITION 20 (1989) (stating that software industry in United States is burgeoning). In recent years the software industry has experienced an annualized revenue growth rate in excess of 30%, and revenues have approached $8 billion dollars annually. *Id.* Customers worldwide range from governments and corporations to individuals using personal home computers, and industry participants range from large corporations to individuals composing programs on personal home computers. *Id.* at 19.

47. *See id.* at 20 (stating that art of writing programs has fostered vital and valuable market both in economic benefits and social benefits). Commonly, the social benefits and economic efficiencies derived from high quality computer programs exceeds the pure financial rewards that a developer receives. *Id.* Computer programs are now at the heart of such everyday activities as the air traffic control system, the bank clearing system, and the payroll system. *Id.*

48. *See id.* at 22 (discussing attributes of software industry and effect on program development). The prominent attributes of the software industry include: low capital requirements, high reliance on skilled intellectual labor, great variability in the skill levels of participants active in industry, and a shortage of extremely good program writers. *Id.* The low capital requirements produce extensive competition because skilled individuals can enter the market with few economic contributions other than their time and energy. *Id.* A result is that a single author or a small group of authors can produce extraordinarily successful programs, and the market has witnessed one time garage shop programmers become major players in the industry. *Id.*

49. *See* The Economist, July 7, 1990, at 69, col. 3 (noting that cheap look-alike versions of well known programs are common).

50. *See supra* note 4 (listing series of cases that deal with issue of copyright protection for computer programs).

51. *See* Apple Computer v. Franklin Computer, 714 F.2d 1240, 1246-47 (3d Cir. 1983) (protecting both source and object code), *cert. dismissed*, 464 U.S. 1033 (1984).

protection beyond a computer program's literal elements, however, has been highly contested, and the resulting court decisions have produced little accord and much confusion.[52] The "nonliteral" elements of a computer program are those elements of a computer program other than the object code and the source code.[53] The nonliteral elements include the "structure, sequence, and organization" of the source code and the object code, the structure of the "user interface" that permits the user to communicate both data and commands to the program, and the presentation of information on the screen.[54] These elements basically encompass those portions of the program that the design phase produces.[55] The continuing debate and confusion over copyright protection of a computer program's nonliteral elements presents a very important dilemma for programmers and manufacturers alike: copyright law may or may not protect the most substantial part of a computer program investment, the design of the program.

One of the first courts to address the copyrightability of a computer program's nonliteral elements was the Third Circuit Court of Appeals in *Whelan Associates v. Jaslow Dental Laboratory*.[56] The *Whelan* court addressed whether the scope of copyright protection extended to a computer program's structure, sequence, and organization or extended only to a computer program's literal code.[57] In addressing this issue the court recognized that the first step was to determine if the structure, sequence, and organization of the computer program represented copyrightable material.[58] The court sought to develop a test that would both facilitate the distinction between idea and expression in computer programs and force a court to balance the competing goals of the copyright law.[59] In developing this test, the court initially classified the computer program in question as a utilitarian

---

52. *See infra* notes 56-163 and accompanying text (discussing different approaches and results of several courts).

53. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 46 (D. Mass. 1990) (stating that nonliteral elements differ from program's written code).

54. *See id.* (stating that nonliteral elements include overall organization of program, structure of program's command system, and presentation of information on screen).

55. *See supra* notes 9-25 and accompanying text (describing development of computer programs).

56. 797 F.2d 1222 (3d Cir. 1986).

57. Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1224 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987). The copyrightability of the structure, sequence, and organization of a computer program was a question of first impression for the Third Circuit in *Whelan*. *Id.* In addressing the question of copyrightability the *Whelan* court stated that the court first must determine whether copyright protection extends beyond a program's literal code. *Id.*

58. *Id.*

59. *Id.* at 1235. The *Whelan* court stated that distinguishing the expression of an idea from the idea itself is often difficult. *Id.* However, the *Whelan* court sought to formulate a rule to facilitate the distinction between expression and idea in the context of computer programs by reviewing relevant copyright precedents. *Id.*

work rather than a fictional or artistic work.[60] Then, relying on *Baker v. Selden*,[61] the court stated that the idea of a utilitarian work is the purpose or function of the work, and, thus, the first step in the test is to determine the purpose or function of the work.[62] Next, the court stated that the expression of the idea is everything that is not necessary to the purpose or function of the work, and, thus, the second step of the test is to determine the aspects of the work that are not necessary to the purpose or function of the work.[63] In determining what is expression under the second step of the test, the court stated that if numerous ways of achieving the desired purpose or function exist, then the specific way chosen is not necessary to the purpose or function and, thus, is protectible expression rather than unprotectible idea.[64]

The *Whelan* court further held that its test does not violate the mandates of the Copyright Act.[65] First, because a court must determine that numerous ways of achieving the desired purpose or function exist, the test does not permit protection of expressions that are limited in number.[66] Second, the test properly balances the ideals of copyright law because the test affords protection to a programmer's most valuable efforts while not retarding the development of new programs that accomplish the same task in a different manner.[67]

In applying their test, the *Whelan* court held that the idea of the allegedly infringed program was to improve the efficiency of the business operations of a dental office.[68] The court then held that the expression of this idea was the manner in which the programmer structured the program to operate and control the computer in receiving, manipulating, and producing useful information.[69] Having separated the program's idea from the program's expression, the court determined that the expression embodied in the program structure was not essential to the idea of the program

---

60. *Id.* at 1238. The *Whelan* court held that the rule which the *Whelan* court developed functions best in the analysis of utilitarian works. *Id.*

61. 101 U.S. 99 (1879).

62. Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1236 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987). The *Whelan* court held that a court may draw the line between a program's idea and the program's expression of the idea by focusing on the end that the program seeks to achieve. *Id.*

63. *Id.*

64. *Id.*

65. *Id.* at 1237. The *Whelan* court held that their test would advance the basic purpose underlying the idea or expression distinction by preserving the balance between competition and protection. *Id.*

66. *Id.* at 1236. In determining whether or not an expression is necessary to the achievement of a desired purpose, the *Whelan* court stated that only if several methods of achieving a desired purpose exist can a court determine that any one method is not necessary to achieving the desired purpose. *Id.*

67. *Id.* at 1237.

68. *Id.* at 1238.

69. *Id.* at 1239.

because other programs existed that accomplished the same task but had different program structures.[70] Based on these conclusions, the court extended copyright protection beyond a program's literal elements to the nonliteral elements of structure, sequence, and organization.[71]

A second court to address the protection of a computer program's nonliteral elements was the District Court for the Northern District of California in *Broderbund Software v. Unison World*.[72] The *Broderbund* court addressed whether the copyright protection of a computer program extends to the program's audiovisual displays, or more specifically, the menu screens, the input formats, and the sequencing of screens.[73] The court considered the idea or expression distinction as the threshold issue, but the court did not utilize the *Whelan* test.[74]

The first element of the idea or expression distinction that the court explored was the idea of merger.[75] Under the merger doctrine copyright law will not protect an expression that is indistinguishable from the idea that the expression represents; the expression "merges" with the idea.[76] Though expressed in a slightly different way, the merger doctrine is analogous to the *Whelan* court's statement that an expression is copyrightable only if the expression is but one way of numerous ways to express the idea.[77] From this brief analysis the court proceeded to address whether the menu screens, the input formats, and the screen sequencing in the program at issue were expressions that were indistinguishable from the underlying idea that they were expressing.[78] The court concluded that these audiovisual elements were distinguishable expression because of the existence of one other program on the market that accomplished the same task in a different manner.[79] This one program satisfied the court that many different ways of expressing the ideas embodied in the allegedly infringed program existed.[80]

---

70. *Id.* at 1240.

71. *Id.* at 1248.

72. 648 F. Supp. 1127 (N.D. Cal. 1986).

73. Broderbund Software v. Unison World, 648 F. Supp. 1127, 1131-32 (N.D. Cal. 1986).

74. *See infra* notes 85-88 and accompanying text (explaining *Broderbund* court's condensation of *Whelan* test).

75. *Broderbund*, 648 F. Supp. at 1131.

76. *Id.*; *see also supra* note 41 and accompanying text (discussing concept of merger doctrine and corresponding prohibition of copyright protection for expressions of an idea that are limited in number).

77. *See supra* notes 63-67 and accompanying text (explaining that number of expressions must be numerous to extend copyright protection to any one expression).

78. Broderbund Software v. Unison World, 648 F. Supp. 1127, 1132 (N.D. Cal. 1986) (addressing defendant's argument that any menu driven program enabling users to create cards, banners, and posters will have substantially same user interface).

79. *Id.* Plaintiffs produced "Stickybear Printer" as evidence of a competing program that accomplished substantially the same functions in a very different manner. *Id.*

80. *Id.*; *see also infra* notes 85-88 and accompanying text (explaining fallacy in *Broderbund* court's approach).

To support the extension of copyright protection to a program's audiovisual displays, the court relied on an interpretation of *Whelan*.[81] The court interpreted *Whelan* to stand for the extension of copyright protection beyond a program's literal aspects to the overall structure of a program, including the program's audiovisual displays.[82] This interpretation of *Whelan*, however, is overexpansive because the *Whelan* court did not address the copyrightability of audiovisual screens.[83] The *Whelan* court only addressed the copyrightability of the structure, sequence, and organization of the program code, and the *Whelan* court merely deemed the screen displays to have some probative value as evidence of alleged copying.[84]

Under the *Whelan* test, audiovisual screen displays would receive copyright protection only if they were not necessary to the underlying function of the program and were expressible in numerous ways.[85] However, the *Broderbund* court condensed the *Whelan* two part test into a single test by analyzing function and expression collectively rather than separately.[86] Under the *Broderbund* analysis, if the audiovisual displays are expressible in a number of different ways then logically they are not essential to the purpose or function of the underlying program.[87] This approach clouds the distinction between idea and expression and can produce confusing and erroneous results because this approach focuses the fact-finder's attention on the expression of the idea before the fact-finder formulates the idea.[88]

---

81. *Broderbund*, 648 F. Supp. at 1133.

82. *Id.*

83. *See* Note, *Copyright Protection for Computer Screen Displays*, 72 MINN. L. REV. 1123, 1145 (1988) (explaining that *Whelan* court never suggested that program's copyright extended beyond program code to embrace screen displays). The *Broderbund* court's extension of copyright protection to the structure, sequence, and organization of program's screen displays is without precedent because the *Broderbund* court misinterpreted *Whelan*. *Id.*; *see also* Digital Communications v. Softklone Distrib. Corp., 659 F. Supp. 449, 455 (N.D. Ga. 1987) (holding that *Broderbund* court's reading of *Whelan* was overexpansive and erroneous).

84. *See* Note, *supra* note 83, at 1145 (stating that *Whelan* court dismissed screen displays as only minimally relevant to proving program infringement because entirely different program codes can produce similar screens).

85. *See supra* notes 63-67 and accompanying text (explaining test for distinguishing protectible expression from unprotectible idea).

86. *See* Broderbund Software v. Unison World, 648 F. Supp. 1127, 1132 (N.D. Ca. 1986) (addressing only whether program's ideas are expressible in numerous different ways rather than separating analysis into purpose or function of program and whether or not chosen expression is essential to program's purpose or function).

87. *See id.* at 1134 (holding that artistic and aesthetic considerations and not utilitarian considerations dictated structure, sequence, and layout of audiovisual displays in allegedly infringed program).

88. *See id.* at 1132 (holding that existence of one other program on market proved that numerous ways existed to express underlying idea embodied in both programs). By beginning with the expression rather than the idea underlying the expression, the *Broderbund* court failed to address the concept of merger. *Id.*; *see also supra* notes 75-80 and accompanying text (defining *Broderbund* court's concept of merger).

The existence of one other program does not sufficiently prove that there are numerous ways to express an idea. *See supra* notes 9-30 and accompanying text (discussing development

Subsequently, in *Digital Communications v. Softklone Distributing*[89] a court again addressed the issue of copyright protection for a computer program's audiovisual screens.[90] The *Digital* court recognized *Whelan* as the leading case extending copyright protection beyond a computer program's source code and object code, but the court refused to embrace the *Broderbund* expansion of *Whelan*.[91] In rejecting the *Broderbund* holding that copyright protection of a computer program extended to the program's audiovisual displays, the court held that copyright protection of a computer program's screen displays must extend from a source other than the copyright registration on the computer program.[92] However, the court noted that the computer screens in question had a separate audiovisual copyright registration, and the court held that a valid copyright registration for a computer program's screen displays provides a basis for copyright protection of the screen displays.[93]

Consequently, the court turned to the idea or expression distinction to determine if the particular status screen at issue was copyrightable.[94] The court did not expressly follow the *Whelan* test, but did utilize a similar approach.[95] First, the court defined the idea behind a status screen.[96] The court determined that the idea behind a status screen is the process or manner by which a status screen operates.[97] Having defined the idea, the court next addressed the expression of a status screen.[98] The court determined that the expression of a status screen is the method by which a status screen communicates the idea to the user.[99]

---

of computer programs and importance of satisfying needs of users to commercial success computer programs). In fact, the existence of only one other program on the market strongly indicates that only a limited number of ways exist to express the idea embodied in the programs at issue. *Id.*

89. 659 F. Supp. 449 (N.D. Ga. 1987).

90. *See* Digital Communications v. Softklone Distrib. Corp., 659 F. Supp. 449, 453 (N.D. Ga. 1987) (noting that defendants contest copyrightability of computer program's status screen as audiovisual display).

91. *See id.* at 455 (interpreting *Whelan* as not specifically extending copyright protection to computer program's screen displays). The *Digital* court held that the *Broderbund* court's reading of *Whelan* was overexpansive and erroneous. *Id.*

92. *Id.* at 456.

93. *Id.* The *Digital* court stated that a certificate of copyright registration represents prima facie evidence of the validity of the copyright on the registered work and raises a presumption of copyrightability of the work's subject matter. *Id.* However, the defendant may rebut the presumption of copyrightability by introducing evidence that the work's subject matter is not copyrightable. *Id.*

94. *Id.* at 457.

95. *Id.* at 458. In applying the idea or expression distinction, the *Digital* court realized that the inherent problem is defining the underlying idea of the copyrighted work. *Id.* Once the court conceptualizes the idea, however, then the court can determine whether or not there are various ways of achieving the desired result. *Id.*

96. *Id.*

97. *Id.*

98. *Id.*

99. *Id.*

Having defined the separate idea and expression embodied in a status screen, the court held that certain elements of the status screen were ideas that could not be copyrighted because they only related to how the user conveys commands to the program and how the program conveys the results of those commands to the user.[100] According to the court, the noncopyrightable elements included the following: the use of the status screen to convey information, the use of a menu command system to operate the program, and the particular method in which the user conveyed the commands to the computer.[101] The court, however, distinguished the presentation of these ideas on the screen as expression entitled to copyright protection because the presentation of the information was unrelated to the operation of the program.[102] The court identified the arrangement of the command terms and the specific highlighting of the command terms as copyrightable elements of expression.[103] Finally, the *Digital* court held that the expression of the status screen did not merge with its underlying idea because the author could arrange the status screen in a variety of ways.[104]

In 1987, a ruling by the Copyright Office cast doubt over the *Digital* court's basic holding that copyright law only protected a computer program's audiovisual screens that an author registered under a separate audiovisual copyright.[105] The Copyright Office, after notice and hearing, issued new guidelines for registering computer programs that limited the number of separate copyright registrations to one per program.[106] Depending on whether the predominant screen displays of the program were pictorial or textual, a programmer could obtain either an audiovisual copyright or a program copyright, but a programmer could not obtain both copyrights for the same program.[107] An audiovisual copyright would protect the underlying computer program, and a program copyright would protect the textual screen displays generated by the program.[108]

---

100. *Id.* at 459.

101. *Id.*

102. *Id.*

103. *Id.* The *Digital* court did place one limitation on the copyrightability of a status screen's expression. *Id.* at 460-61. If the status screen merely indicates where to record data, then the status screen is a blank form that copyright law does not protect. *Id.* at 461 (noting Baker v. Selden, 101 U.S. 99 (1879)). If, however, the arrangement of the status screen is sufficiently innovative so that the status screen actually conveys information to the user, then copyright law should protect the status screen. *Id.*

104. *Id.* at 460.

105. *See* Registration Decision, 53 Fed. Reg. 21817, 21818 (1988) (indicating concern that *Digital* decision would cast doubt on copyright protection of computer screen displays where author failed to obtain separate registration on screen displays).

106. *See id.* at 21817 (informing public that Copyright Office considers as one work all copyrightable expression owned by same claimant and embodied in computer program, including computer screen displays, and requires registration on single application form).

107. *See id.* at 21818 (indicating existence of two different copyright registrations). If the authorship of the computer program is predominately literary, then the registrant should use form TX. *Id.* If the authorship of the computer program is predominately audiovisual, then the registrant should use form PA. *Id.*

108. *See id.* (indicating that single registration covers all copyrightable elements embodied

However, decisions by the Copyright Office, though controlling over the registration process, are not binding authority on courts.[109] Therefore, under the Copyright Office's new single copyright registration policy, a programmer who obtains a program copyright registration may not receive any protection for the program's screen displays.[110] Accordingly, a court that chooses to follow the *Digital* opinion will not extend the copyright protection of the underlying program to the program's screen displays.[111]

Shortly after the Copyright Office's ruling, the Connecticut District Court addressed the issue of copyright protection of a computer program extending to the program's screen displays in *Manufacturers Technologies v. Cams.*[112] In *Cams* the district court refused to follow the *Broderbund* decision because the *Broderbund* decision overextended copyright protection by ignoring that more than one program can generate the same screen display.[113] The *Cams* court approved of the *Digital* court's decision not to extend the copyright protection of a computer program beyond that established in *Whelan* while upholding the validity of a second copyright on a program's screen displays.[114] Realizing that the *Digital* decision and the Copyright Office ruling were at odds, the court created the legal fiction of two separate, but interrelated copyright registrations derived from the single copyright registration permitted by the Copyright Office.[115] Under this legal fiction, copyright protection extended to both the program and its screen displays to the extent that each contained copyrightable material.[116]

---

in work). The Copyright Office stated that a registrant should give a description on the application as either "entire work" or "computer program." *Id.* at 21819. This description would cover the program's code and screen displays regardless of whether or not the registrant deposited material identifying the program's screen displays with the Copyright Office. *Id.* If the registrant, however, refers to specific screen displays in the description of the work on the registration form, the registrant must deposit material that visually identifies the screen displays. *Id.* at 21820.

109. *See* Note, *supra* note 83, at 1150 (explaining that in jurisdiction following *Digital* precedent, developer that is unable to register screen displays separately has no legal recourse against competitor that utilizes developer's screen displays but produces them with different computer program code).

110. *See id.* at 1143 (stating that disallowing simultaneous but separate registration of screen displays places developers desiring to protect their screen displays at mercy of courts following *Digital* precedent).

111. *See supra* notes 89-108 and accompanying text (explaining *Digital* court's rationale and effect of Copyright Office ruling).

112. 706 F. Supp. 984 (D. Conn. 1989).

113. *See* Manufacturers Technologies v. Cams, 706 F. Supp. 984, 992-93 (D. Conn. 1989) (explaining that *Broderbund* approach ignores fact that more than one computer program can produce virtually identical screen displays).

114. *See id.* at 993 (stating that computer program and computer program's screen displays are fundamentally distinct). By refusing to extend copyright protection of the underlying program to the program's screen displays, courts can focus on the unique copyrightable elements in each and avoid the mistake of confusing the underlying idea of the program with the underlying idea of a particular screen display. *Id.*

115. *Id.* The *Cams* court held that the legal fiction of two separate copyrights derived

Having established the legal fiction framework, the *Cams* court proceeded with the idea or expression distinction adopted by the *Digital* court.[117] The court, however, faced a different situation than the *Digital* court because a sequence of screen displays was at issue instead of a single status screen.[118] To compensate for this difference, the court divided the screen displays into external and internal aspects.[119] The court deemed the sequence and flow of the screen displays to be the external aspects of the screen displays.[120] The court then classified the placement of the screen heading, the program commands, and the chosen function or selection on each individual screen as the internal aspects of the screen displays.[121]

Consequently, the court held that the external aspects of the program in question were copyrightable because they represented the expression of the programmer's idea of how to best accomplish the process of making a cost estimate.[122] According to expert testimony, the process functionally does not limit expression because each individual creating a cost estimate often uses a specific process unique to that individual.[123] As for the internal aspects of the screen displays, the court held that some aspects were copyrightable and others were not copyrightable.[124] For example, the format of the screens was not copyrightable because the size of the screen physically limited the placement of command menus and status lines to a narrow range of possibilities.[125] Also not copyrightable because of the limited choices available were the method of navigation within each screen and the menu driven navigation between successive screens.[126] The court, however, did

---

from a single copyright registration would permit courts to give copyright protection to the copyrightable elements in both the program and the program's screen displays while conforming to the registration policies of the Copyright Office. *Id.*

116. *Id.*

117. *See id.* at 994 (stating that idea of program was process of creating cost estimate).

118. *See id.* at 993-94 (indicating that programmer registered screen displays in question as "compilations" under § 101 of Copyright Act). Because the programmer registered the works as "compilations," the *Cams* court held that the trial court must perform a bifurcated analysis of the external aspects and internal aspects of the screen displays. *Id.* at 994.

119. *Id.* at 994.

120. *Id.*

121. *Id.* at 994-95.

122. *See id.* at 994 (stating that making cost estimate was primary function of program). The programmer in *Cams* designed the program in question to enable the user to estimate the cost of machining a manufactured part without having to perform manual calculations. *Id.* at 988. Making a cost estimate involves synthesizing information concerning selection of materials, assignment of material costs, selection of number of operations, selection of type of operations, and assignment of engineering costs, sales costs, and handling costs. *Id.*

123. *See id.* (explaining that cost estimate process is unique to each individual). Each individual will usually prioritize, select, and arrange the operations to be performed in a different way. *Id.*

124. *See id.* at 995-98 (describing internal elements that are copyrightable and internal elements that are not copyrightable).

125. *Id.* at 995. The physical size of the screen limits the placement of components on the screen to some type of vertical or horizontal configuration. *Id.*

126. *Id.* at 995-96.

hold that the composition of the status screen, the arrangement of the specific terms within the menus, the assignment of certain terms to certain operations, and the construction of a status report represented copyrightable expression.[127]

The most recent case involving copyright protection of computer programs is *Lotus Development Corp. v. Paperback Software International.*[128] The key issue in *Lotus* was whether the nonliteral elements of a computer program are copyrightable, and if they are copyrightable, how a court could distinguish these nonliteral elements from those nonliteral elements that are not copyrightable.[129] To answer this question, the *Lotus* court first examined the Copyright Act and the Act's underlying object and policy to determine how best to draw the line between the copyrightable and the noncopyrightable elements of a computer program.[130] The court concluded that the idea or expression distinction remained the best vehicle for this determination because the distinction forces courts to weigh the competing goals of the copyright law in making a decision.[131] Drawing the line between the literal and nonliteral elements of a computer program was too restrictive and, furthermore, did not permit the courts properly to consider the goal of copyright law to encourage creativity and to make ideas available for public benefit by granting authors a limited monopoly.[132]

To address properly the idea or expression distinction the court considered four critical concepts: originality, functionality, obviousness, and merger.[133] First, the court stated that an expression is copyrightable only if the expression is original and only if the expression does not embody elements of the idea that are functional in the utilitarian sense.[134] Second, the court stated that if an expression is obvious or is expressible in only a limited number of ways, then copyright protection does not extend to the expression because the expression merges with the idea and is inseparable from the idea itself.[135] Consequently, recognizing that no established legal test existed that encompassed these concepts in a factual situation involving a computer program copyright, the court created a three pronged legal test to govern issues of computer program copyright protection.[136]

---

127. *Id.* at 996.

128. 740 F. Supp. 37 (D. Mass. 1990).

129. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 46 (D. Mass. 1990) (describing amorphous nature of nonliteral elements of computer program).

130. *See id.* 47-53 (detailing Congressional mandates embodied in Copyright Act and relevant judicial interpretations of Copyright Act).

131. *See id.* at 54 (concluding that Congress manifested intention to use idea or expression distinction as part of test of copyrightability for computer programs).

132. *See id.* (concluding that idea or expression distinction is consistent with object and policies of copyright).

133. *Id.* at 58-59.

134. *Id.* at 58.

135. *Id.* at 58-59. The *Lotus* court stated that the concept of merger was only a slight extension of the "obviousness" doctrine. *Id.* at 59. The difference is that under the merger doctrine the expression is separable from the idea, but the available expressions are limited in number. *Id.*

136. *See id.* at 59 (stating that no copyright statute or previous copyright cases adequately

Under the *Lotus* court's legal test a court first must make some formulation or conception to define the idea of the program in question.[137] This formulation must be done on a linear scale that runs from a generalized conception of the idea to a more particularized conception of the idea.[138] Although a bright line rule to follow in developing these formulations is nonexistent, attempting to define the idea represented by the computer program from a general level to a more specific level will provide the court with a good starting point.[139] Once the court is satisfied that the formulation of the program's idea is sufficient, the second step is to analyze the program's expression of the idea.[140] Specifically, the court must determine if the expression is limited to elements essential to the expression of the idea or, alternatively, if the expression is one of only a limited number of ways to express the idea.[141] Finally, if the court determines that elements are present which are not essential to every expression of the idea and are expressible in numerous ways, then the court must decide if the unique elements are a substantial part of the allegedly copyrightable work.[142]

---

defined legal test for determining copyrightability of computer program's nonliteral elements); *infra* notes 137-142 and accompanying text (defining elements of proposed three pronged legal test on computer copyright); *see also* Comment, *Copyright Protection for Computer Software after* Whelan Associates v. Jaslow Dental Laboratory, 54 Mo. L. REV. 121, 132 (1989) (stating that development of a proper test for computer program copyright infringement remains unsettled issue).

137. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 60 (D. Mass. 1990) (holding that formulation of idea is necessary for later distinguishing expression of idea from idea itself).

138. *See id.* (incorporating abstractions test established by Learned Hand in Nichols v. Universal Pictures Corp., 45 F.2d 119, 121 (2d Cir. 1930), *cert. denied*, 282 U.S. 902 (1931)); In *Nichols*, Nichols authored and copyrighted the play "Abie's Irish Rose." *Nichols*, 45 F.2d at 120. Subsequently, Universal Pictures produced a motion picture play entitled "The Cohens and the Kellys," and Nichols alleged that Universal Pictures copied his expression embodied in "Abie's Irish Rose." *Id.* In addressing this allegation, Judge Learned Hand stated that for a play, or any given work, as a fact finder methodically omits specific details of the play, the idea and expression that the play embodies become more and more general. *Id.* At the most general level, the title of a play may represent what the play is about. *Id.* at 121. However, as a fact finder unveils these layers of increasing generality in a series of abstractions, the fact finder reaches a point where copyright can no longer protect the particular expression of the idea. *Id.* To hold otherwise would permit a playwright to "prevent the use of his 'ideas,' to which, apart from their expression, his property is never extended. Nobody has ever been able to fix that boundary, and nobody ever can." *Id.*; *see also infra* notes 151-153 and accompanying text (describing *Lotus* court's application of idea formulation process).

139. *See Lotus*, 740 F. Supp. at 60-61 (stating that placing idea of program along scale of abstraction is not precisely calculable).

140. *See id.* at 61 (stating that identifying elements of expression of idea is critical to remainder of analysis).

141. *See id.* (defining second prong of legal test as analyzing expression of program's idea).

142. *See id.* (defining third prong of legal test as determining if expression is substantial part of copyrighted program). In determining substantiality the *Lotus* court advocates a qualitative analysis as well as a quantitative analysis. *Id.*; *see also* SAS Inst.v. S & H Computer Sys., 605 F. Supp. 816, 829-30 (M.D. Tenn. 1985) (stating that question of substantiality is also question of fact). A quantitatively small part of a copyrighted work may represent a qualitatively substantial part of the copyrighted work. *Id.*

The *Lotus* court emphasized that this is not a bright line test, and, thus, the decisionmaker must evaluate several relevant factors.[143] The court defined these factors as the relevant characteristics of the copyrighted work, the relevant characteristics of the allegedly infringing work, and the relevant circumstances of their development and use.[144] By advocating that a decisionmaker consider these factors, the court is emphasizing that a decisionmaker should not focus solely on the copyrighted work in proceeding through the legal test.[145] By focusing on both programs present in the infringement suit, the decisionmaker can reach a more sound and informed conclusion as to the copyright protection afforded the allegedly infringed program.[146]

Satisfaction of this test merely establishes that the allegedly infringed work is copyrightable.[147] The reviewing court must proceed to the remaining issues involved in a suit for infringement, including whether the allegedly infringing work actually infringed the copyrighted work.[148] The *Lotus* court stated that a benefit of this legal test is that the test reduces the potential confusion of the decisionmaker by more sharply focusing the issue being adjudicated.[149] The reason is that the test encourages the advocates involved to moderate their extreme positions on the scope of copyright protection of computer programs in favor of more supportable positions that best address the issue involved.[150]

---

143. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 61 (D. Mass. 1990) (explaining inherent nature of legal tests that requires weighing of factors or elements).

144. *Id.*

145. *Id.*

146. *Id.*

147. *See id.* (stating that satisfying test establishes copyrightability of allegedly infringed work).

148. *See id.* (stating remaining issues of infringement suit). Other remaining issues include whether or not the plaintiff perfected the copyright on the copyrighted work, or proved any damages because of the infringement. *Id.*

149. *See id.* at 62 (describing how *Lotus* test will aide court by focusing issues for adjudication).

150. *See id.* (describing how *Lotus* test will effect positions that advocates adopt in arguing for and against copyright protection for computer programs). The advocate for the copyright owner will encourage the court to adopt a very generalized idea of the copyrighted work. *Id.* If the court defines the idea generally, then there will be many possible expressions of the idea. *Id.* The result is that the copyright owner's expression will be one among many, and the court will extend copyright protection to the copyright owner's expression. *Id.* The advocate for the owner of the allegedly infringing work will encourage the court to adopt a very particularized idea of the copyrighted work. *Id.* If the court defines the idea very specifically, then the court's definition will limit the number of available expressions of the idea. *Id.* The result is that the court will hold that the copyright owner's expression is one of a limited number of possible expressions, and, therefore, not entitled to copyright protection. *Id.* However, if either advocate advances too extreme a position, the idea will lose credibility when the debate forces the advocate to propose a less extreme and more sensible position at a later time. *Id.*

Having established a framework for analysis, the court proceeded to address the particular facts of Lotus' copyright infringement claim.[151] The court stated that the general idea of the two programs was the idea of an electronic spreadsheet.[152] On a more specific level, the court defined the idea of the two programs as an electronic spreadsheet that utilized a two-line moving cursor as a user interface.[153]

The court next addressed the specific elements of the expression embodied in the Lotus program.[154] Under this analysis the court determined that several aspects of the Lotus expression were not copyrightable because of limited expression and merger.[155] These elements included the rotated "L" screen display, the keys used to invoke the command menus and move the cursor, and the symbols used to perform mathematic calculations.[156] The court, however, extended copyright protection to the structure, sequence, and organization of the command menu system.[157] In doing this, the court emphasized the distinction between the idea of a menu command system and the expression of a menu command structure.[158] While a menu command system is not copyrightable as an idea, the structural format of a menu command structure is expressible in many different ways.[159] Based

---

151. *Id.*

152. *See id.* at 65 (stating that idea of electronic spreadsheet is not copyrightable because it is functional and obvious).

153. *See id.* (stating that two line moving cursor menu is functional and obvious). The *Lotus* court noted that many different programs utilize a two-line moving cursor menu, and copyright protection did not extend to the idea of a two-line moving cursor menu. *Id.* However, the *Lotus* court stated that copyright protection might extend to a particular expression of a two-line moving cursor menu if the expression included original and distinctive elements beyond those essential to stating the idea itself. *Id.*

154. *See id.* (stating that issue is whether Lotus program goes beyond those details essential to any expression of electronic spreadsheet and includes substantial elements of expression that are distinctive and original and, thus, copyrightable).

155. *See id.* at 66-67 (discussing expressive aspects of Lotus program that copyright does not protect); *supra* notes 41-45 and accompanying text (discussing limited expression and merger doctrine).

156. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 66 (D. Mass. 1990) (describing elements of expression not entitled to copyright protection).

157. *See id.* at 67 (stating that structure, sequence, and organization of menu command system may contain distinctive elements of expression not necessary to every expression of electronic spreadsheet idea (citing Whelan Assoc. v. Jaslow Dental Laboratory, 797 F.2d 1222, 1248 (3d Cir. 1986), *cert. denied*, 479 U.S. 1031 (1987)). The *Lotus* court stated that a particular expression of a menu structure is not essential to electronic spreadsheet idea. *Id.* Furthermore, the *Lotus* court stated that a particular expression of a menu structure does not merge with a less abstract idea of a menu structure for an electronic spreadsheet because the idea of a menu structure is expressible in an unlimited number of ways. *Id.*

158. *See id.* (describing menu command structure as expression of menu command system idea).

159. *See id.* (comparing menu command structures of competing electronic spreadsheet programs and describing specific elements subject to different expression). Although some of the specific command terms "merge" with the idea of the command term, this merger does not preclude copyrightability as a "compilation" under the Copyright Act for the command structure taken as a whole. *Id.*

on this analysis, the court held that the Lotus menu command structure was an original and nonobvious way of expressing a command structure that fell within the protection of copyright law.[160]

Having decided that copyright law protects the specific menu command structure of Lotus, the court proceeded to the "substantial aspect" portion of the legal test to determine if copyright protection should extend to the Lotus command structure.[161] The court stated that this portion of the test answered itself under the circumstances because the menu command structure is Lotus' most unique and important aspect.[162] With all three prongs of the legal test sufficiently addressed and answered, the court concluded that copyright law protected the menu command structure.[163]

The *Lotus* decision is the latest in a long line of computer program copyright cases that have produced little accord in this unsettled area of law.[164] Because of recent trends in the computer software industry, the need for a common framework of analysis never has been greater.[165] For example, after the *Lotus* ruling, Lotus immediately filed copyright infringement suits against two other competitors.[166] Software companies now view lawsuits as a tool not only to protect their products, but also to bolster their balance sheets by raising much needed cash.[167] Thus, courts must resolve computer copyright law by devising a consistent test for computer copyright protection.

Despite the lack of an appellate affirmation,[168] the *Lotus* decision represents a comprehensive and workable framework for deciding computer program copyright infringement cases.[169] The first prong of the *Lotus* test

---

160. *Id.* at 68.

161. *Id.*

162. *Id.*

163. *Id.*

164. *See supra* notes 56-163 and accompanying text (discussing present confusion in area of copyright protection for computer programs).

165. *See* Daily Telegraph, Aug. 26, 1990, at 22, col. 1 (stating that legal battles over intellectual property will proliferate in computer programming industry).

166. *See* MACWEEK, July 10, 1990, at 113 (stating that immediately after *Lotus* ruling, Lotus filed infringement suit against Borland International and Santa Cruz Operation).

167. *See* Richards, *Firms Cash In On Patent Cases*, Wash. Post, Oct. 21, 1990, at H1, col. 1 (stating that drive to cash in on intellectual property is picking up steam as country's one-time technological powerhouses find it harder and harder to compete against Asian conglomerates and nimble start-ups at home).

168. *See* Alexander, *Lotus Development to Get $500,000 in Copyright Case*, Wall St. J., Oct. 18, 1990, at B4, col. 4 (reporting that following *Lotus* decision Lotus and Paperback Software settled out of court). As part of the settlement, Paperback Software agreed not to appeal the *Lotus* decision. *Id.*; *see also* Ould, *Experts See Further Turbulence as Copyright Law Establishes Itself*, PC WEEK, July 16, 1990, at 116 (stating that *Lotus* decision is not binding on other courts). Without an affirmation on appeal, other courts may discount the precedential value of the *Lotus* ruling and continue to develop their own analytical frameworks. *Id.*

169. *See* Antton and Hoffman, *Copyright Protection and Innovation:* The Impact of Lotus Development v. Paperback Software, 7 THE COMPUTER LAWYER, Aug. 1990, 1, 4 (stating that three part test of *Lotus* should prove far easier to apply in practice than overly broad, conclusive dictum of *Whelan*).

focuses the fact finder's attention on the idea underlying the allegedly infringed program; defining the underlying idea of the program is the logical place to start this analysis.[170] Although conceptualizing the idea may be difficult, the process will provide insight into the various ways to express the idea and into the overall complexity of the idea.[171] Once the court sufficiently formulates the idea, the second prong of the *Lotus* test directs the court to focus the analysis on the particular expression of the idea embodied in the program.[172] This shift in focus automatically brings the goals of copyright law into play because the fact finder must weigh the originality of the author's expression against the number of available expressions of the idea.[173] However, the *Lotus* court warned that, under this second prong, a reviewing court should not resort to a "look and feel" analysis to determine the copyrightability of the elements of a computer program.[174] The third prong of the *Lotus* test forces the court to limit the extension of copyright protection only to expressions that represent substantial portions of the computer program.[175] This limitation on the extension of copyright protection is beneficial to the software industry because programmers can focus on significant innovations while not constantly worrying

---

170. *See supra* notes 137-139 and accompanying text (describing first prong of *Lotus* test).

171. *Id.*

172. *See supra* notes 140-141 and accompanying text (describing second prong of *Lotus* test).

173. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 58 (D. Mass. 1990) (stating that second prong of *Lotus* test forces court to weigh competing goals of copyright law).

174. *See id.* at 62-63 (explaining why courts should avoid doctrine of "look and feel" as distinguishing between copyrightable and noncopyrightable elements).

The "look and feel" doctrine is a recent theory advanced by copyright registrants in computer copyright infringement cases that purports to supplant the traditional claim that a competitor has infringed a program copyright by copying the program's expression of an idea. *See* L. KUTTEN, *supra* note 2, § 2.03[4][a] (stating that in increasing number of cases copyright owners are claiming that alleged infringer copied "look and feel" of registered program). Under the "look and feel" approach, a copyright owner files suit alleging that the infringer has violated his copyright by copying the program's user interface to such an extent that a typical program user cannot distinguish between the two programs. *Id.* In defense, the infringers claim that the "look and feel" of the registered program represents only the idea of the program and, thus, copyright does not protect the "look and feel" of the program. *Id.* However, despite the new packaging, the "look and feel" doctrine still encompasses the basic elements of idea and expression, and counsel should still apply the traditional idea or expression distinction in advancing copyright infringement claims. *Id.*

The *Lotus* court rejected the "look and feel" doctrine as a basis for distinguishing between the copyrightable and noncopyrightable nonliteral elements of a computer program. *Lotus*, 740 F. Supp. at 58. Courts should only use the look and feel doctrine to determine whether or not improper copying has occurred. *Id.* If courts use the "look and feel" doctrine to determine copyrightability, the courts may assume that because two programs look alike, copyright law extends to the similar portions of the copyrighted program. *Id.* The look and feel doctrine, consequently, forces courts to reach a conclusion of copyrightability without addressing the reasons for extending copyright protection under the competing goals of the copyright law. *Id.*

175. *See supra* note 142 and accompanying text (describing third prong of *Lotus* test).

about infringing relatively insignificant portions of another author's work.[176]

Industry participants may feel uneasy with the *Lotus* test because they cannot easily determine the portions of a program that copyright law protects and does not protect.[177] This difficulty of application, however, may work in favor of industry competitors because the *Lotus* test provides more narrow copyright protection to any given computer program by forcing courts to define the idea of the program beyond the most general level.[178] If a court defines the idea of a computer program very generally, then the range of protectible expressions becomes very broad.[179] As the court defines the idea of a computer program more specifically, the range of protectible expressions necessarily narrows.[180]

A comparison of the *Whelan* test and the *Lotus* test illustrates that the range of protectible expression narrows as the courts define the idea of a computer program more specifically.[181] Under the *Whelan* test the idea of a computer program is the program's purpose or function, and the expression of a computer program's idea is everything that is not necessary to the program's purpose or function.[182] To satisfy the *Whelan* test a court only needs to determine the general purpose or function of the computer program.[183] Once the court defines the purpose or function of the computer program, the court immediately proceeds to determine if the alleged expressions of the program's idea are necessary to the program's purpose or function.[184] The court then extends copyright protection to any expression that is not necessary to the purpose or function of the computer program

---

176. *Id.*

177. *See supra* notes 137-144 and accompanying text (describing that *Lotus* test is not bright line test); *see also* Petraske, *An Infringement Test for Comprehensive Similarity in Software Cases*, 7 THE COMPUTER LAWYER, Aug. 1990, 12, 21 (stating that for copyright protection test to be of any value, competitor must be able to apply test to decide what is protectible and what is not protected). Although difficult, the *Lotus* test is at least of some help in identifying that which is protected from that which is available for copying. *Id.*

178. *See* Petraske, *supra* note 177, at 20-21 (explaining that more general definition of idea creates broader range of copyrightable expression).

179. *Id.*

180. *Id.*

181. *See supra* notes 178-180 and accompanying text (describing that range of protectible expressions of computer program idea narrows as court defines idea of computer program more specifically).

182. *See supra* notes 59-64 and accompanying text (describing *Whelan* test for determining copyrightability of computer programs).

183. *See* NIMMER ON COPYRIGHT § 13.03 [F] at 13-62.27-62.28 (1990) (stating that *Whelan* court failed to realize that more than one idea may underlie general purpose or function of computer program). Nimmer stated that "[t]he *Whelan* court eschewed Judge Hand's advice and adopted a simplistic test for separating idea from expression in cases involving computer programs. . . . The crucial flaw in this reasoning is that it assumes that only one idea, in copyright law terms, underlies any computer program, and that once a separable idea can be identified, everything else must be expression." *Id.*

184. *See supra* notes 59-64 and accompanying text (describing *Whelan* test for determining copyrightability of computer programs).

as long as the expression does not merge with the idea because of a limited number of available expressions.[185]

If the *Lotus* court had followed the *Whelan* test without modification, the *Lotus* court would have provided the Lotus program with much greater protection.[186] The *Lotus* court initially defined the idea of the Lotus program as an electronic spreadsheet.[187] Under the *Whelan* test, the *Lotus* court would have stopped defining the idea of the Lotus program at this level and addressed the expressions embodied in the Lotus program.[188] The *Lotus* court would have focused on the two-line moving cursor menu as an expression of the electronic spreadsheet idea embodied in the Lotus program.[189] As an expression, the two line moving cursor menu is not essential to the purpose or function of the Lotus program because many ways of expressing a menu exist.[190] Consequently, under the *Whelan* test, the *Lotus* court would have ruled that copyright protection extended to the Lotus program's expression of a two-line moving cursor menu.[191] This holding would have been detrimental because such a holding would prohibit any computer programmer from using a two-line moving cursor menu in a competing electronic spreadsheet program.[192]

Nevertheless, some observers have criticized the *Lotus* decision as extending copyright protection to an element of the Lotus program that is inherently utilitarian and, therefore, generally unprotectible.[193] The end user

---

185. *Id.*

186. *See infra* notes 187-192 and accompanying text (discussing how *Whelan* test would provide Lotus program with greater copyright protection than *Lotus* test).

187. *See* Lotus v. Paperback Software, 740 F. Supp. 37, 65 (D. Mass. 1990) (stating that at most general level computer programs under consideration are expressions of idea of computer program for electronic spreadsheet).

188. *See supra* note 183 and accompanying text (stating that *Whelan* test assumes that only one idea underlies general purpose or function of computer program).

189. *See supra* note 152 and accompanying text (stating that electronic spreadsheet is most general idea underlying Lotus program).

190. *See Lotus*, 740 F. Supp. at 67 (describing varied forms menu screen may assume). Some of the many forms that a menu screen may assume include a one line menu, a three line menu, a four line menu, a pull down or vertical menu, and a command driven interface. *Id.*

191. *See supra* notes 63-64 and accompanying text (discussing that under *Whelan* test if numerous ways of achieving purpose or function of computer program exist, specific way chosen is not necessary to purpose or function of computer program and, thus, is protectible expression).

192. *See supra* notes 31-45 and accompanying text (describing elements of copyright infringement suit and importance of idea or expression distinction in determining copyright protection of register work). Ownership of a valid copyright registration raises a rebuttable presumption that copyright protection subsists in the registered work. *Id.* The owner of an allegedly infringing work can rebut this presumption by proving that copyright law does not protect the allegedly infringed portions of the copyrighted work. *Id.* If the *Lotus* court held that the two-line moving cursor menu in the Lotus program was protectible expression, then no competing programmer could utilize the two-line moving cursor menu and survive a copyright infringement suit. *Id.*

193. *See* Abramson, *Why Lotus - Paperback Uses the Wrong Test and What the New*

uses the command menu interface to interact with the computer program, and, consequently, the user interface provides utility to the computer program.[194] Section 102(b) of the Copyright Act denies copyright protection of utilitarian elements except where artistic elements are separable from the utilitarian aspect.[195] Although copyright law does not protect a utilitarian element, copyright law does protect a programmer's expression of a utilitarian element.[196] A court may have difficulty distinguishing the unprotectible utilitarian elements of a computer program from the protectible expressions of the utilitarian elements,[197] but the *Lotus* decision illustrates that the formulation of a good idea will facilitate this distinction.[198] Under the *Lotus* facts, the Lotus programmer used the two-line moving cursor menu as the process or method to communicate with the user.[199] The particular Lotus command structure within the two-line moving cursor menu is the programmer's expression of the process or method embodied in the two-line moving cursor menu.[200]

Observers also consider the first part of the *Lotus* test to be susceptible to error because the test provides no guidelines for a court to know at what level of specificity to stop defining the idea of a program.[201] This vulnerability disturbs observers because this portion of the test effects the remainder of the court's analysis.[202] Although observers may consider this vulnerability detrimental, Congress wisely has mandated this vulnerability by refusing to

---

Software Protection Legislation Should Look Like, 7 THE COMPUTER LAWYER, Aug. 1990, 6 (stating that *Lotus* decision threatens to upset many widely-held beliefs and related practices in computer and software industries).

194. *See id.* at 7 (stating that interactive elements of program and other results produced by operation of program have utility).

195. *See* Copyright Act, *supra* note 1, § 102(b) (describing limitations on copyright protection for utilitarian elements).

196. *See* HOUSE REPORT, *supra* note 3, at 5670 (expanding on legislative intent underlying Copyright Act § 102(b)). The HOUSE REPORT states that "[s]ection 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." *Id.*

197. *See* Abramson, *supra* note 193, at 7 (arguing that separating expressive aspects from utilitarian aspects of software often is made more difficult by fact that utility of computer program results largely from communication of information to and from user). The expression of a computer program is largely a result of the communication of information to and from the end user. *Id.* Accordingly, a fact finder possibly may characterize certain communication as expression and also may determine that the communication is an inseparable part of the program's utilitarian function. *Id.*

198. *See supra* notes 151-160 and accompanying text (discussing *Lotus* court's application of *Lotus* test to Lotus electronic spreadsheet).

199. *Id.*

200. *Id.*

201. *See supra* notes 137-139 and accompanying text (discussing that no bright line rules exist for defining idea of program under first prong of *Lotus* test).

202. *See infra* notes 205-207 and accompanying text (describing how formulation of idea as first step of *Lotus* test can adversely affect remainder of analysis).

create a bright line test for copyright protection and instead placing properly in the hands of the courts the task of balancing the competing goals of copyright law.[203] By permitting the courts to consider the applicable copyright protection for a computer program on an ad hoc basis, Congress has ensured that copyright law will remain flexible enough to protect both present and future technological developments in the computer software industry.[204]

The *Lotus* test forces a court to determine the idea of the computer program as the very first step, and this determination greatly effects the results in the remaining two steps.[205] However, computer programs often are complex, and the end user often cannot discern this underlying complexity.[206] If the court cannot sufficiently formulate the program's idea, then the court cannot sufficiently separate the copyrightable expression from the idea, and, consequently, the entire test will fail.[207] To formulate the idea of a computer program, the *Lotus* test currently permits the court to consider the suggestions of counsel and anticipates that the suggestions of counsel will more sharply focus the issues.[208] If the court cannot sufficiently formulate the idea of the computer program with the aid of counsel, the *Lotus* test should permit the court to admit expert testimony to clarify the issue.[209] If the court still cannot sufficiently formulate the idea of the program, then the court should hold that copyright law does not protect the nonliteral elements of the program because the court cannot separate the idea and the expression of the program.[210]

Although the *Lotus* test attempts to force courts to apply the merger doctrine when determining the copyrightability of computer program expressions, courts consistently have applied the merger doctrine improperly.[211] The courts usually have failed to properly apply the merger doctrine when

---

203. *See supra* notes 37-45 and accompanying text (describing Congressional mandates underlying Copyright Act).

204. *Id.*

205. *See* Abramson, *supra* note 193, at 6 (stating that court draws line between expression and idea as very first step of *Lotus* test analysis). Once a court establishes the idea of the computer program, the remainder of the *Lotus* test analysis flows directly from this determination. *Id.*

206. *See id.* at 7 (stating that underlying complexity of well constructed software system is often nearly invisible to end user).

207. *See id.* (stating that inevitably fact finder will formulate idea too broadly and *Lotus* test will fail). The analysis fails because when the fact finder formulates the idea too broadly, the remainder of the analysis characterizes anything else of significance as expression. *Id.*

208. *See supra* notes 147-150 and accompanying text (describing benefits of *Lotus* test in permitting counsel to assist in formulating idea of program).

209. *Id.* Courts may want to admit expert testimony at this early stage of the trial in every instance because the formulation of the program's idea is essential to the remainder of the analysis. *Id.* Without expert testimony the court may not know if the court properly has formulated the idea of the program. *Id.*

210. *See supra* note 40 and accompanying text (discussing lack of copyright protection for expressions that are inseparable from idea they express).

211. *See supra* note 41 and accompanying text (discussing merger doctrine).

other programs exist that compete with the allegedly infringed program, but the other manufacturers are not named defendants in the infringement suit.[212] The courts rely on the existence of these competing programs to establish that other methods are available to accomplish the same function, but in different and original ways.[213] Therefore, the courts rule that the particular expression embodied in the program at issue is entitled to copyright protection.[214] The fallacy of this analogy is that the existence of other programs that accomplish the same function in a different manner merely indicates that other forms of expression exist.[215] The existence of these other programs alone does not prove that an unlimited number of other expressions are available.[216] However, by following this analogy, reviewing courts conclude that the number of available expressions is unlimited without actually addressing whether or not only a limited number of possible expressions actually exist.[217] Therefore, the issue remains as to how many different expressions must be available before "limited number" is no longer a concern.

One consideration is that although a programmer may choose from an unlimited number of expressions to accomplish a desired function, the forces of the marketplace may limit the commercially viable expressions.[218] The reason is that "any expression" will not suffice because the expression must be one that the programmer's audience will accept.[219] An extremely successful program may further decrease the number of commercially acceptable expressions by setting a standard that computer users will use to judge all other programs.[220] Denying copyright protection to a program's expression because the program is successful violates the copyright goal of encouraging the generation of new ideas that will improve the public welfare.[221] If, however, protection of the expression retards innovation, the protection offends the copyright goal of not allowing protection that is detrimental to the best interests of the public.[222]

Courts must resolve this conflict by balancing the goals of copyright protection to reach some form of compromise.[223] The *Lotus* test represents

---

212. *See supra* notes 78-80 and accompanying text (discussing how *Broderbund* court relied on market presence of one competing computer program to determine that many ways existed to express idea embodied in allegedly infringed program).

213. *Id.*

214. *Id.*

215. *Id.*

216. *Id.*

217. *Id.*

218. *See supra* notes 10-11 and accompanying text (discussing increasing importance of understanding user's needs in designing program that users will accept).

219. *Id.*

220. *Id.*

221. *See supra* notes 37-45 and accompanying text (describing Congressional mandates underlying Copyright Act).

222. *Id.*

223. *Id.*

a workable compromise to resolve the copyright conflict between the merger doctrine and the limited availability of commercially viable expressions. The *Lotus* test provides the most comprehensive framework for the copyright analysis of a computer program because the *Lotus* court based the test on a strict reading of the Congressional mandates embodied in the Copyright Act.[224] Although courts in other jurisdictions do not have to follow the *Lotus* decision as precedent, courts in other jurisdictions should follow the *Lotus* test. As other courts follow and apply the *Lotus* test, the courts will develop a comprehensive and coherent body of law pertaining to the copyrightability of computer programs. This body of law will then provide the software industry with better guidelines to judge the copyrightability of their future works contrasted with the confusion that still exists today.

Clark H. Worthy

224. *See* Lotus Dev. Corp. v. Paperback Software Int'l, 740 F. Supp. 37, 47 (D. Mass. 1990) (stating that court must examine relevant language of copyright statute, provisions of whole law, and object and policy of copyright statute in determining if copyright protection extends to nonliteral elements of computer program).